

AM65x/DRA80xM EMIF Tools

ABSTRACT

At the center of every application is the need for memory. With limited on-chip processor memory, external memory serves as a solution for large software systems and data storage, and an unstable external memory interface can result in system failures or hinder software development. To prevent potential system level anomalies and ensure robust systems, hardware must be configured correctly and tested thoroughly.

The EMIF Tools application focuses on post layout activities, including configuring the Texas Instruments' processors for accessing external double data rate (DDR) memories and optimizing delay lines to compensate for skew. This application report provides a detailed description on how to use the associated application files. The document overview provides a complete list of processors and memory types supported by the EMIF Tools application.

The spreadsheet discussed in this application report can be downloaded [here](#).

Refer to the [AM65x/DRA80xM DDR Board Design and Layout Guidelines](#) application note for information on supported designs using this EMIF Tools application.

Contents

| | | |
|---|---|---|
| 1 | Overview | 2 |
| | 1.1 Supported Features of EMIF Tools..... | 2 |
| 2 | EMIF Configuration | 2 |
| | 2.1 Preliminary Requirements | 2 |
| | 2.2 Save/Load Configuration Files | 2 |
| | 2.3 Generating EMIF Register Values..... | 2 |

List of Tables

| | | |
|---|--|---|
| 1 | EMIF Register Configuration Worksheets | 2 |
|---|--|---|

1 Overview

This document provides detailed steps outlining the procedure to initialize Texas Instruments' processors to access external DDR memories using the accompanying tools included in the EMIF Tools application.

1.1 Supported Features of EMIF Tools

The EMIF Tools application supports the following features:

- TI SOCs: AM65x, DRA80xM
- DDR Types: DDR3L, DDR4, LPDDR4
- Basic EMIF configuration topics including:
 - Initializing the EMIF and DDR for basic read/write functionality
 - Compensating for signal skew
 - Enabling the error correction code (ECC) feature of the EMIF interface

The source code generated in the EMIF tool includes a GEL file that can be run on the R5 or the A53 cores of the supported TI SOCs. The tool also generates a .dtsi file that can be built with u-boot to facilitate DDR configuration in a Linux environment.

2 EMIF Configuration

The following section describes how to use the supporting application files to configure the EMIF controller for DDR memory accesses.

2.1 Preliminary Requirements

Before using the supporting application files, ensure that you have access to the data sheet of the selected DDR memory.

2.2 Save/Load Configuration Files

The first worksheet provides two buttons that lets the user save/load DDR Tool configurations. The Save Configuration button saves the configuration parameters from the spreadsheet into a text file. It uses certain parameters from the configuration as the default filename; however, any filename can be used. The Load Configuration button can read a saved configuration generated by the Save Configuration button, and load the configuration into the appropriate cells in the spreadsheet. The text files can be read if needed, but should not be modified, as the buttons use Excel macros that expect a certain format.

2.3 Generating EMIF Register Values

To assist you in defining the EMIF configuration register values, the EMIF Tools application provides an EMIF register configuration workbook. The workbook is divided into six worksheets, and requires specific information pertaining to the system application environment. The worksheets with tabs highlighted in yellow requires your input. The worksheets with tabs highlighted in green are automatically generated, and are used to configure the EMIF controller.

Table 1. EMIF Register Configuration Worksheets

| Worksheet | Description |
|----------------------|---|
| Step1-SystemDetails | User input: system information |
| Step2-DDRTimings | User input: timing requirements from DDR datasheet |
| Step3-AddressMapping | User input: setup address map of the DDR controller |
| Registers | Output: calculated register values |
| GEL | Output: GEL formatted file to be used during initialization |
| uboot | Output: uboot .dtsi file to be used in Linux environment |

The following sections outline the procedure to complete the various required input parameters of the EMIF configuration workbook.

2.3.1 Step1 – System Details

The first worksheet requires you to input both high level system application details, as well as specific I/O settings for the DDR pins of the TI application processor and DDR memory.

Step 1A seeks system level details and is described below.

The parameters are defined in detail in the bulleted list below:

- **DDR Memory Type:** This value should be selected based on the DDR memory type.
- **DDR Memory Frequency:** This value should be selected for the desired DDR clock frequency.
- **DDR bus width (single device):** This is the width of a single package. For example, a 32-bit wide DDR interface consisting of 4 DDR devices would enter a value of 8.
- **DDR Data Bus Width:** This value should be selected based on the bus width between the TI processor and the DDR memory. This value represents the total data bus width including all of the attached memories (not including ECC byte). Pre-defined values are provided in a drop-down menu list.
- **Enable ECC:** Setting this parameter to “Yes” enables the ECC functionality of the EMIF controller.

Care should be taken to provide details in order throughout the workbook. As the parameter values are selected, the drop-down menu lists of other parameters may change.

Step 1B requests specific details pertaining to the DDR memory used in the system application. These details are required for the workbook to determine the size and speed bin of the DDR memory.

The parameters are defined in detail in the bulleted list below:

- **Speed Bin (Data Rate):** The data rate at which the connected DDR memory operates. A drop down menu provides supported data rates. This is not the data rate the memory is capable of (which is typically higher), but rather the maximum rate at which the memory operates.
- **Density:** This value should be selected to match the density of a single DDR memory connected to the TI processor.
- **Number of Rows:** Enter the number of rows of the DDR memory.
- **Number of Columns:** Enter the number of columns of the DDR memory.

Step 1C requires the user to provide the desired I/O settings for the DDR memory termination and output driver impedance.

Change the cells in yellow to select the appropriate value in the drop down menu. The shaded cells will change automatically. Recommended values are provided as examples from TI evaluation boards. Board-level simulations and signal integrity analysis should be performed to ensure the appropriate settings for your specific board design.

The parameters are defined in detail in the bulleted list below:

- **ODT/Rtt_Nom:** This value applies to the on-die termination of the DDR memory I/O pins.
- **Dynamic ODT / Rtt_Wr:** This value applies to on-die termination during DDR writes when the dynamic ODT mode is enabled.
- **Output Driver Impedance:** This value applies to the output driver impedance of the DDR memory I/O pins.

Step 1D allows you to modify the I/O settings for the DDR pins of the TI application processor.

Change the cells in yellow to select the appropriate value in the drop down menu. The shaded cells will change automatically. Recommended values are provided as examples from TI evaluation boards. Board-level simulations and signal integrity analysis should be performed to ensure the appropriate settings for your specific board design.

The parameters are defined in detail in the bulleted list below:

- **ODT / Rtt:** This value applies to the on-die termination of the DDR I/O pins on the Texas Instruments application processor.
- **Output Driver Impedance (Addr/Ctrl/Clk):** This value applies to the output driver impedance of the DDR

address, control, and clock I/O pins on the Texas Instruments application processor.

- Output Driver Impedance (Data/Strobe): This value applies to the output driver impedance of the DDR data and strobe I/O pins on the Texas Instruments application processor.

2.3.2 Step 2 – DDR Timing

This worksheet requires you to input DDR timing values that can be found in the DDR memory data sheet. Here are some tips in filling out this worksheet:

- The worksheet includes support for all three DDR types supported by the TI processor. Some rows may not be applicable to your DDR type. The Comments column provides clues on which rows apply to your device. If a row does not apply to your device, leave the associated yellow cells blank, or fill with zero.
- DDR data sheets provide the timing parameters either in clock cycles (tCK) or ns, or sometimes the maximum of both. Fill out either or both parameters that are provided in the DDR data sheet. If a value doesn't exist in the DDR data sheet, leave it blank, or fill with zero. The Final Datasheet value column calculates the value in nanoseconds. This value is used in calculating values in the Registers tab.
- Some cells are white. These are automatically calculated based on input from other values. Do not adjust these cells.
- The Parameter field values change depending on the DDR type. These values should correspond to the same name as in the DDR data sheet.
- Some values depend on the speed bin of the device. Use the values for the speed bin of the device you are using, or the speed at which you are operating the device.
- Some values depend on the page size and device density. Ensure that you choose the correct parameters when filling out the worksheet.
- All listed parameters require minimum timing value, unless otherwise noted by '(max)'
- Ensure the values are correct when inputting parameters from the data sheet. There is no data validation check based on expected JEDEC parameters. This will be included in a future revision of the tool.

2.3.3 Step3 Address Mapping

This worksheet provides a method to individually map address signals (row, column, and bank) to different internal address signals. This allows the user to customize how the controller addresses the memory to potentially optimize accesses to/from the memory for a particular application.

The worksheet automatically selects a default memory map based on the DDR type you have chosen. Ensure the default mapping corresponds to the number of rows, columns, and banks you have in your memory device. Address bits that are not used should be chosen as "NA". Use the drop down menus to choose each address bit, as the selections are limited based on the DDR type. Also, the worksheet checks for duplicates. The cell highlights in green if the selection does not have a duplicate. All non-NA cells should be green when completed.

The default selection for LPDDR4 and DDR3L has the column bits in the lower bits of the address map, followed by the bank bits, followed by the row bits. Similarly for DDR4, column bits are in the lower bits of the address map, followed by 2 banks bits and 2 bank group bits, followed by all the row bits. This is the recommended default address mapping.

For 16-bit designs, the most significant column bit should not be used in the address mapping, and all other bits shifted down. For example, if your DDR has 10 column address bits, only use col0-8 in the address mapping sheet if you only have a 16-bit data bus.

Do not leave any holes in the address mapping (that is, all NA cells should be in the upper most bits).

2.3.4 Registers

After the System Level, DDR timing, and Address Mapping details have been populated in their corresponding worksheets, you can access the Registers worksheet. This contains most of the calculated values needed to generate the source code output of the tool. This worksheet is for information purposes only. Refer to this to find how individual bit fields are programmed based on the inputs provided. Further worksheets are provided which provide the full configuration needed for the controller and PHY registers.

2.3.5 GEL

The GEL tab provides the output to be used in a GEL configuration script for the DDR controller and PHY configuration. The button "Export GEL" lets you save the file in the correct format to be used by the GEL initialization scripts provided by TI to initialize the processor's DDR controller and PHY. The button generates a file called M4_DDR_config.gel, which can be directly used in the suite of GEL files needed for customized initialization of the DDR.

2.3.6 u-boot

The u-boot tab provides the output to be used in a Linux build for DDR controller and PHY configuration. It is compatible with the Linux Processor SDK for the AM65x. The "Export u-boot .dtsi" button lets you save the file in the correct format to be used in u-boot, to initialize the processor's DDR controller and PHY. The button generates a source file *.dtsi, which can be directly used in your u-boot build. Typically, the file .dtsi can be placed in arch/arm/dts in u-boot and can be included in k3-am654-r5-base-board.dts.

2.3.7 RTOS DDR Initialization

The RTOS tab provides the output to be used in a RTOS build for DDR controller and PHY configuration. It is compatible with the RTOS Processor SDK for the AM65x. The "Export RTOS .h header file" button lets you save the file in the correct format to be used in the RTOS PDK to initialize the processor's DDR controller and PHY. The button generates a source file board_ddr_config.h, which can be directly used in your RTOS build. Typically, the file can be placed in pdk_am65xx_x_x_x/packages/ti/board/src/evmKeystone3/include to be built into the RTOS PDK.

Revision History

NOTE: Page numbers for previous revisions may differ from page numbers in the current version.

| Changes from A Revision (November 2018) to B Revision | Page |
|--|-------------|
| • Updated u-boot section. | 5 |
| • Added RTOS DDR Initialization section. | 5 |

IMPORTANT NOTICE AND DISCLAIMER

TI PROVIDES TECHNICAL AND RELIABILITY DATA (INCLUDING DATASHEETS), DESIGN RESOURCES (INCLUDING REFERENCE DESIGNS), APPLICATION OR OTHER DESIGN ADVICE, WEB TOOLS, SAFETY INFORMATION, AND OTHER RESOURCES "AS IS" AND WITH ALL FAULTS, AND DISCLAIMS ALL WARRANTIES, EXPRESS AND IMPLIED, INCLUDING WITHOUT LIMITATION ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NON-INFRINGEMENT OF THIRD PARTY INTELLECTUAL PROPERTY RIGHTS.

These resources are intended for skilled developers designing with TI products. You are solely responsible for (1) selecting the appropriate TI products for your application, (2) designing, validating and testing your application, and (3) ensuring your application meets applicable standards, and any other safety, security, or other requirements. These resources are subject to change without notice. TI grants you permission to use these resources only for development of an application that uses the TI products described in the resource. Other reproduction and display of these resources is prohibited. No license is granted to any other TI intellectual property right or to any third party intellectual property right. TI disclaims responsibility for, and you will fully indemnify TI and its representatives against, any claims, damages, costs, losses, and liabilities arising out of your use of these resources.

TI's products are provided subject to TI's Terms of Sale (www.ti.com/legal/termsofsale.html) or other applicable terms available either on ti.com or provided in conjunction with such TI products. TI's provision of these resources does not expand or otherwise alter TI's applicable warranties or warranty disclaimers for TI products.

Mailing Address: Texas Instruments, Post Office Box 655303, Dallas, Texas 75265
Copyright © 2020, Texas Instruments Incorporated