Configuring Cache Controller and DDR Controller in U-Boot Running on SmartFusion2 Starter Kit

Table of Contents

Purpose

This application note describes how to configure Cache Controller and DDR Controller for SmartFusion2® system-on-chip (SoC) field programmable gate array (FPGA) Starter Kit running U-boot.

Introduction

The SmartFusion2 SoC FPGA devices integrate fourth generation flash-based FPGA fabric, an ARM® Cortex™-M3 processor, and high performance communication interfaces on a single chip. The high speed memory controllers in the SmartFusion2 SoC FPGA devices are used to interface with the external double data rate (DDR)2/DDR3/low power DDR (LPDDR) memories. The Cortex-M3 processor can directly run the instructions from external DDR memory through the microcontroller subsystem (MSS) DDR (MDDR).

The SmartFusion2 devices integrate an 8 KB instruction cache. The SmartFusion2 SoC FPGA Cache Controller and MDDR bridge handles the data flow for a better performance. Refer to the *SmartFusion2 Microcontroller Subsystem User Guide* for more information on the Cache Controller. Figure 1 shows the top level block diagram of the reference design running U-Boot.

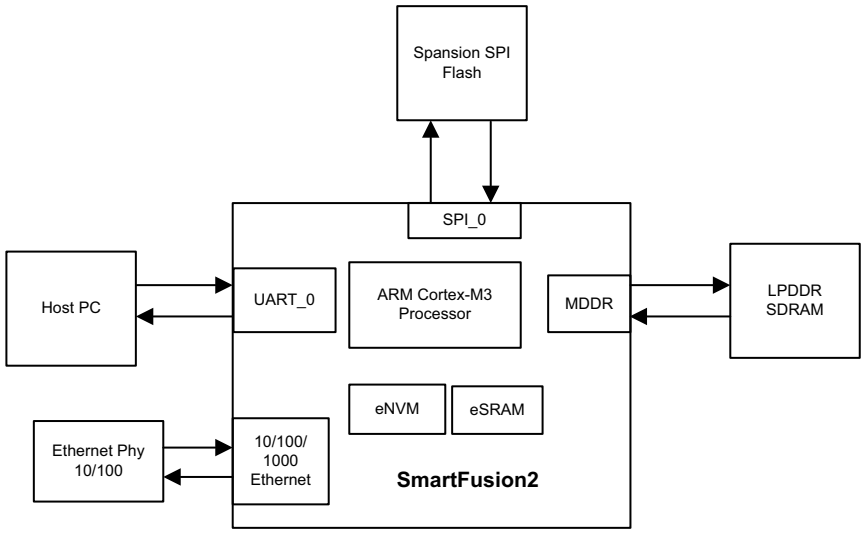

***Figure 1* • Top Level Block Diagram of SmartFusion2 Starter Kit**

U-Boot is a multi-functional open-source bootloader that allows the developer to either load an Operating System from a variety of devices or configure low-level platform. On reset, SmartFusion2 runs U-Boot firmware from the on-chip eNVM/eSRAM. U-Boot performs SmartFusion2 initialization from power-on/reset, including setting up the DDR Controller, enabling cache, and enabling access to the SPI Flash. U-Boot copies the Linux bootable image from the SPI Flash to the LPDDR and then, passes control to the kernel entry point.

Design Description

The Libero design uses MMUART_0, SPI_0, Ethernet, eNVM, eSRAM, DDR, and Cache Controllers. Refer to *SmartFusion2 SOM Hardware Architecture Specification* for more information on the hardware architecture of the Emcraft Systems SmartFusion2 System-On-Module.

Libero design files for SmartFusion2 Starter kit are provided in "Appendix A: Design Files" on page 11.

Hardware Description

Figure 2 shows the Libero SoC SmartDesign.

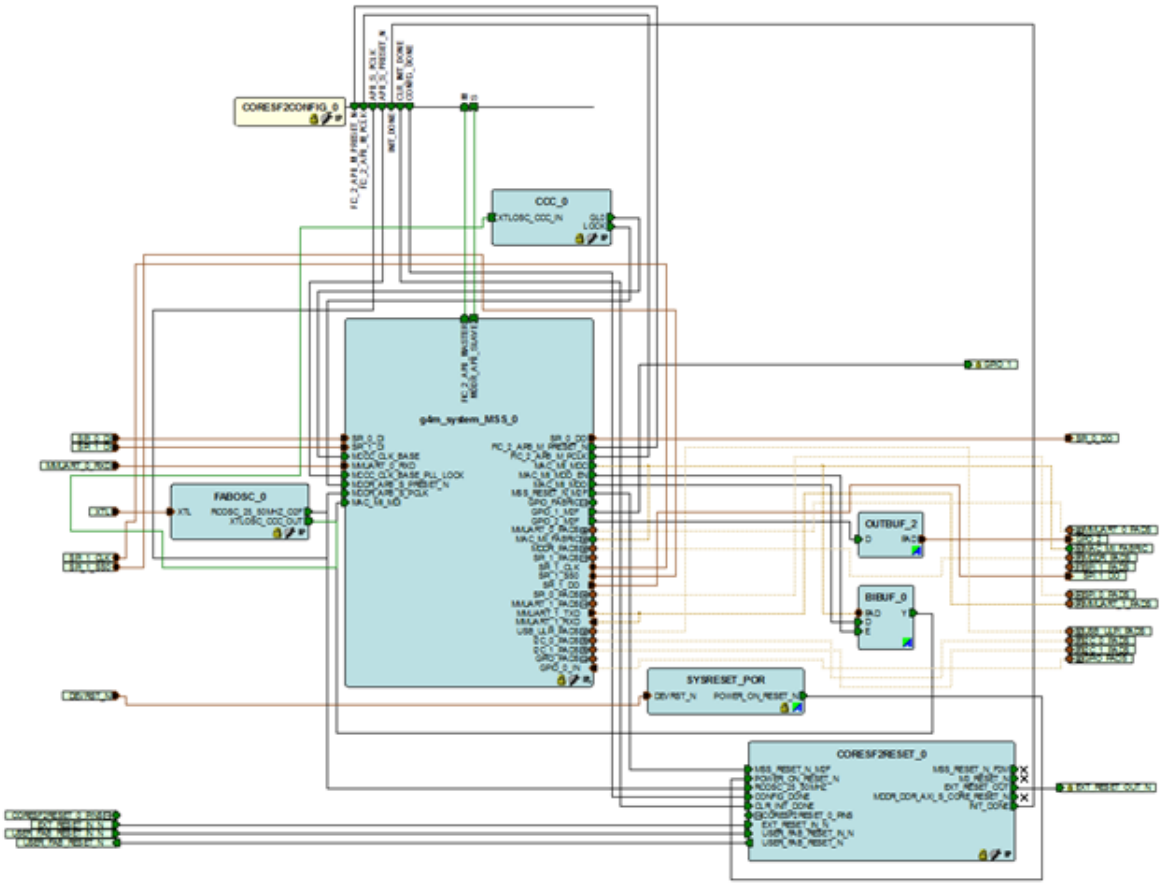

***Figure 2* • SmartFusion2 Starter Kit Hardware Design**

Libero hardware project uses the following SmartFusion2 MSS resources:

- MII Interface to Ethernet PHY
- SPI_0 for reading from/writing to SPI flash
- MDDR for LPDDR SDRAM interface
- UART_0 as a console interface for the U-Boot and Linux software. UART_0 TX and RX are connected to the mini-B USB through the fabric and fabric I/Os.

The cache controller can be configured either using cache controller block in MSS configurator or through system registers.

Figure 3 shows the configuration of cache controller using MSS configurator in the Libero.

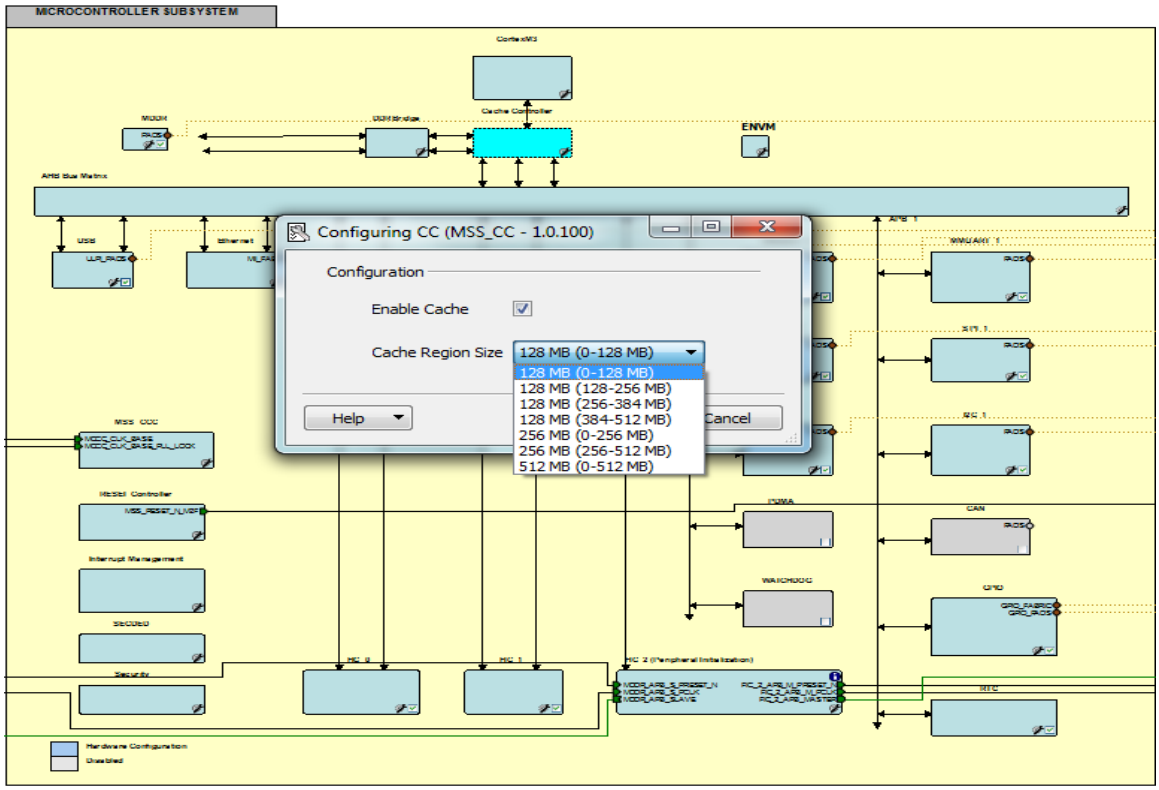

***Figure 3* • Cache Controller Configuration using MSS Configurator**

Software Description

U-Boot source files for SmartFusion2 Starter Kit are provided in"Appendix A: Design Files" on page 11. The U-Boot installation directory displays as:

```
u-boot/api
u-boot/board
u-boot/common
u-boot/cpu
u-boot/disk
u-boot/doc
u-boot/driver
u-boot/examples
u-boot/fs
u-boot/include
u-boot/lib_arm
u-boot/lib_avr32
u-boot/lib_blackfin
u-boot/lib_generic
u-boot/lib_i386
u-boot/lib_m68k
u-boot/lib_microblaze
```

u-boot/lib_mips
u-boot/lib_nios
u-boot/lib_nios2
u-boot/lib_ppc
u-boot/lib_sh
u-boot/lib_sparc
u-boot/libfdt
u-boot/nand_spl
u-boot/net
u-boot/onenand_ipl
u-boot/post
u-boot/tools

Directories of U-Boot for SmartFusion2 Starter Kit

The following are the directories used for porting the U-Boot for SmartFusion2 Starter Kit:

- include\
- board\
- cpu\
- lib_arm\

The include Directory

The include directory contains all the header files used globally by U-Boot. The directoryhas the following files to support SmartFusion2 Starter Kit:

- configs\m2s-som.h
- asm-arm\arch-m2s\m2s.h
- asm-arm\arch-m2s\ddr.h

The configs\m2s-som.h file describes SmartFusion2 MSS features (SPI, UART, Ethernet, Cache Controller, and so on) to be enabled, memory layout configuration, drivers to include, and settings.

The asm-arm\arch-m2s\m2s.h file contains processor-specific definitions required for U-Boot to access the CPU.

The asm-arm\arch-m2s\ddr.h file defines the SmartFusion2 external memory controller configuration registers required for U-Boot to initialize the DDR before loading the Operating System or any application to LPDDR memory.

The board Directory

The board directory contains the board.c file in emcraft\m2s-som sub-directory. This file contains the LPDDR chip parameter and timing definitions. It contains the following board-specific initializations:

- DRAM initialization
- Ethernet initialization

The cpu Directory

The cpu\arm_cortexm3 directory contains initialization routines for internal peripherals of SmartFusion2. These include:

- Cache initialization
- Clock initialization
- Timer initialization

The `arm_cortexm3` directory contains `u-boot.lds` file that defines memory used by the U-Boot. This file defines the starting address and size of the following sections of the U-Boot firmware after building:

- Text
- Data
- Bss
- Stack

Text section of U-Boot is mapped to the eNVM (virtual view) of the SmartFusion2 starting at 0x00000000. Data, Bss, and Stack sections are mapped to eSRAM memory space starting at 0x20000000. Refer to *Emcraft Systems SmartFusion2 SOM Starter Kit Guide* for building the U-Boot firmware.

The `lib_arm` Directory

The `lib_arm` directory contains the `board.c` file that performs the following functions:

- Cache initialization
- Clock initialization
- Serial communication initialization
- DDR initialization

DDR Configuration

The SmartFusion2 is connected to the 64 MBytes of 16-bit LPDDR SDRAM through high-speed memory controller. LPDDR is mapped to the DDR_SPACE 0 (0xA000_0000 to 0xAFFF_FFFF) of M3 data region. When the SmartFusion2 SOM board is reset, U-Boot firmware starts executing from on-chip eNVM. U-Boot configures the DDR Controller before loading the Linux image from SPI flash to LPDDR.

Macros

The following macros are defined in the `include\configs\m2s-som.h` file:

- **CONFIG_NR_DRAM_BANKS**: Number of DDR2/DDR3/LPDDR banks
- **CONFIG_SYS_RAM_BASE**: Starting address of DDR2/DDR3/LPDDR
- **CONFIG_SYS_RAM_SIZE**: Size of DDR2/DDR3/LPDDR memory

The following DDR parameter and timing macros are defined in `board\emcraft\m2s-som\board.c` file. The macro values in `board.c` file are specific to LPDDR SDRAM (MT46H32M16LFBF-6) in the SmartFusion2 Starter Kit. These macros should be updated based on the DDR2/DDR3/LPDDR memories.

- **DDR_BL**
- **DDR_MR_BL**
- **DDR_BT**
- **DDR_CL**
- **DDR_WL**
- **DDR_tMRD**
- **DDR_tWTR**
- **DDR_tXP**
- **DDR_tCKE**
- **DDR_tRFC**
- **DDR_tREFI**
- **DDR_tCKE_pre**
- **DDR_tCKE_post**
- **DDR_tRCD**
- **DDR_tRRD**
- **DDR_tRP**
- **DDR_tRC**
- **DDR_tRAS_max**
- **DDR_tRAS_min**

- **DDR_tWR**

Refer to `board\emcraft\m2s-som\board.c` file for complete details.

Global Variables

DDR Configuration Registers are declared in `asm-arm\arch-m2s\ddr.h` file. This includes DDR Controller Registers, DDR PHY Configuration Registers, and FIC-64 registers.

Refer to `asm-arm\arch-m2s\ddr.h` file for complete structure of DDR Configuration Registers.

Functions

DDR initialization function is defined in `board\emcraft\m2s-som\board.c` file.

The syntax is `int dram_init (void)`.

This function is included in the initialization sequence. On every reset, this function gets executed, if **CONFIG_NR_DRAM_BANKS** macro is greater than zero.

Refer to `board\emcraft\m2s-som\board.c` file for complete function definition.

Cache Configuration

SmartFusion2 devices integrate 8 KB of instruction cache. The Cache Controller caches the program instructions only. The Cache Controller can treat eSRAM, eNVM, and external DDR memory as cacheable memory. To treat any of these memories cacheable, it needs to be remapped to Cortex-M3 code region from 0x00000000 to 0x1FFFFFFF.

Figure 4 shows high-level memory mapping of Cortex-M3.

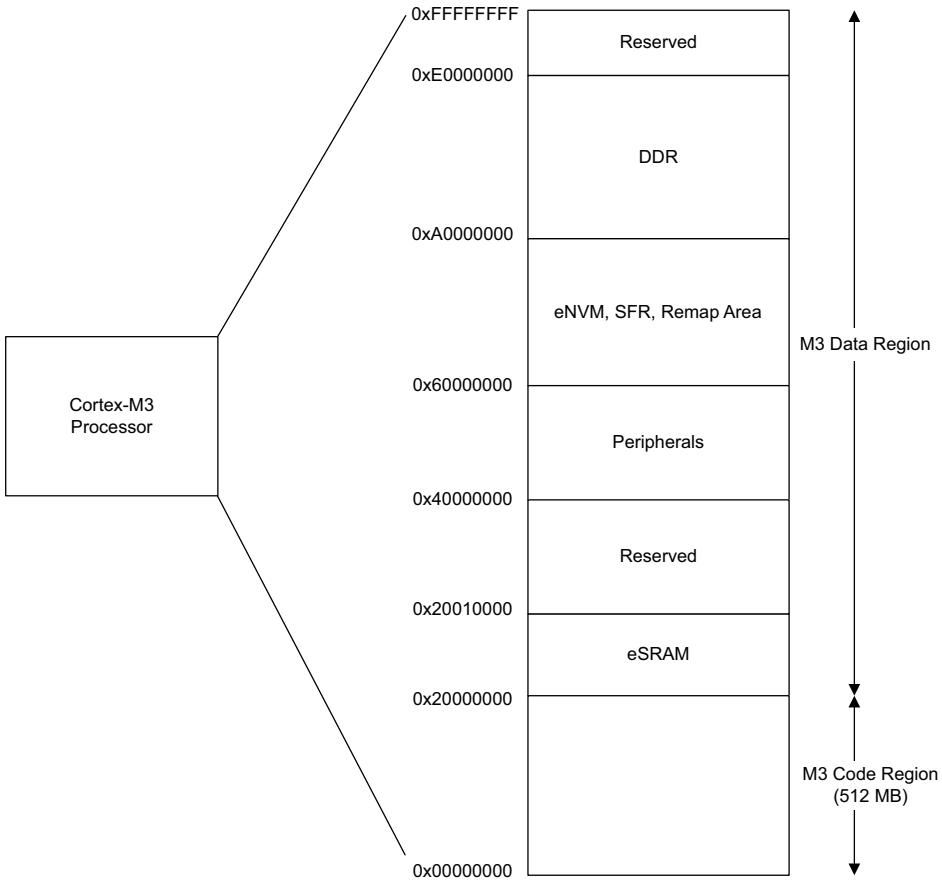

***Figure 4* • SmartFusion2 Cortex-M3 Memory Mapping**

By default, the full eNVM memory of 512 KB (from 0x60000000 to 0x6007FFFF) is mapped as a cacheable region. The eNVM base address 0x60000000 is remapped to Cortex-M3 processor address

space of 0x00000000. It is possible to remap DDR or SDRAM memory address to the bottom (0x0000_0000) of the Cortex-M3 processor code region by using the DDR_CR System Register, and any portion of the mapped memory can be made as cacheable. The cacheable region can be configured to 128 MB, 256 MB, or 512 MB, by using CC_REGION_CR System Register.

Figure 5 shows the system-level view of the Cache Controller in SmartFusion2 SoC FPGA devices.

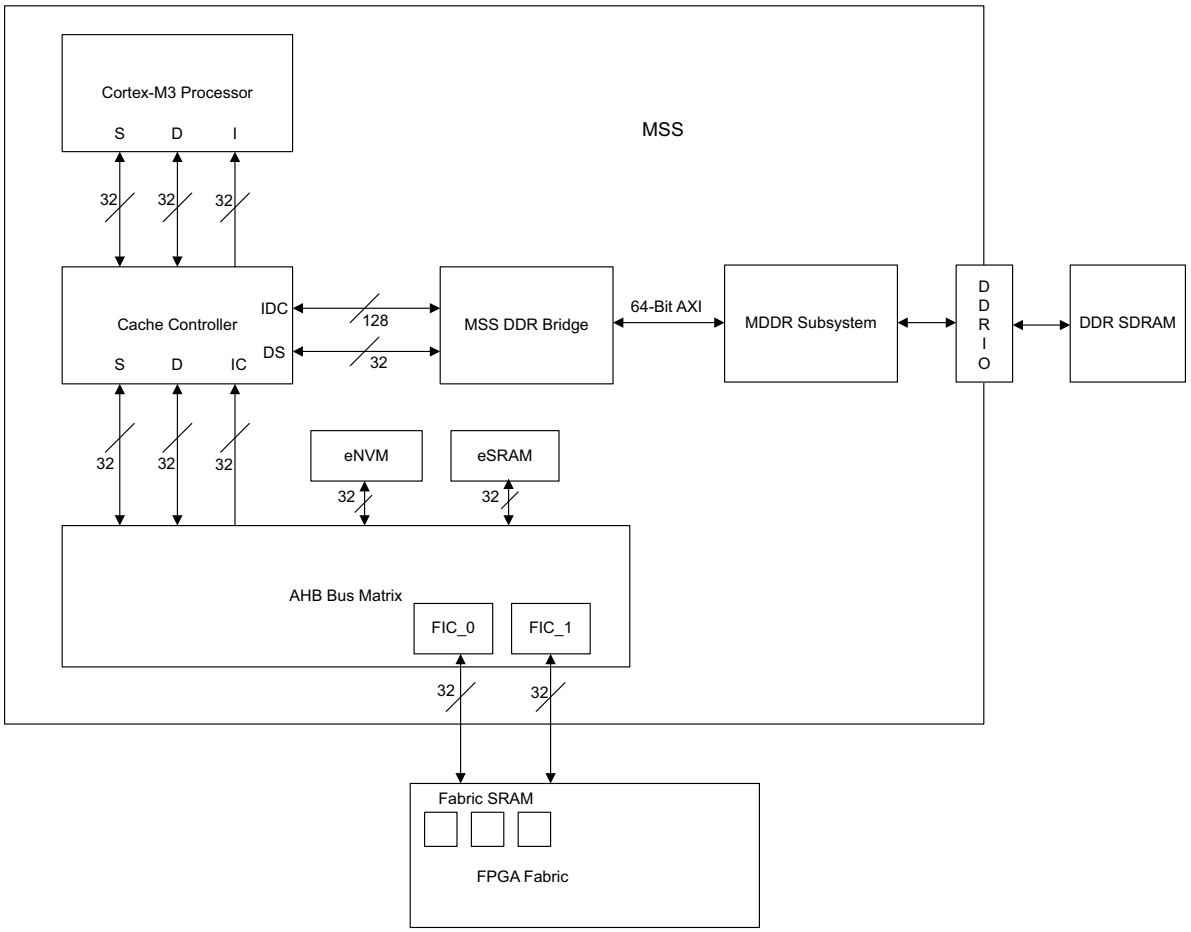

***Figure 5* • System-Level View of Cache Controller in SmartFusion2 SoC FPGA Devices**

Macros

The `u-boot\include\configs` folder contains the header file `m2s-som.h`. This file defines the macro to enable the cache: **CONFIG_M2S_CACHE_ON**.

Global Variables

SmartFusion2 System Registers are used to configure the Cache Controller. System Registers are declared in `include\asm-arm\arch-m2s\m2s.h` file.

Refer to `include\asm-arm\arch-m2s\m2s.h` file for complete structure of System Registers.

Functions

Cache initialization function is defined in `cpu\arm_cortexm3\m2s\soc.c` file.

The syntax is `static void soc_cache_enable(void).`

On every reset, this function gets executed, if the **CONFIG_M2S_CACHE_ON** macro is defined. This function configures the cache System Registers. Refer to `cpu\arm_cortexm3\m2s\soc.c` file for complete details.

Caching in U-Boot

U-Boot firmware is loaded to the eNVM of the SmartFusion2 using the Libero eNVM client as shown Figure 6. By default, the full eNVM memory (from 0x60000000 to 0x6007FFFF) is mapped as a cacheable region. The eNVM base address 0x60000000 is remapped to Cortex-M3 processor address space of 0x00000000, which is Cortex-M3 code region. The complete U-Boot firmware is in cacheable region (from 0x00000000 to 0x1FFFFFFF) of the Cortex-M3.

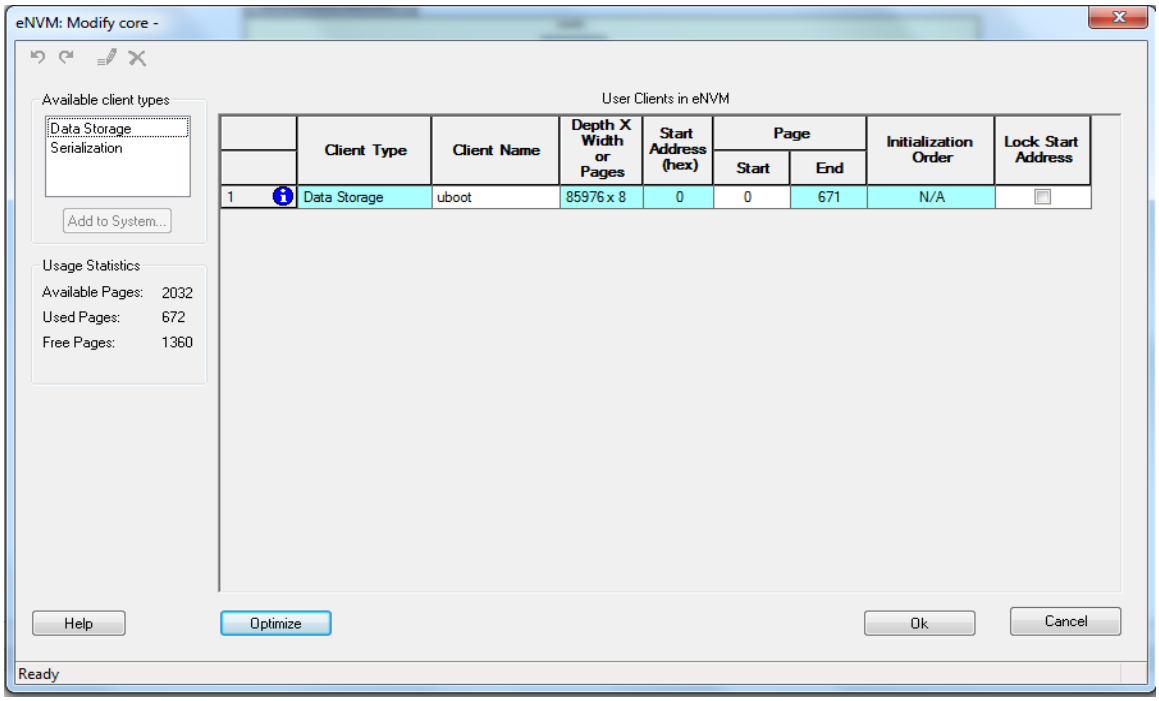

***Figure 6* • Adding the U-boot Using the Libero eNVM Client**

Caching in Linux Kernel

U-Boot copies the Linux bootable image from the SPI flash to the LPDDR and then passes control to the kernel entry point. SmartFusion2 Emcraft Starter Kit has 64 MB of LPDDR memory. LPDDR memory is interfaced through DDR interface (from 0xA0000000 to 0xDFFFFFFF) of 1 GB for both code and data regions. LPDDR uses a total of 25 address lines (13 row address lines, 10 column address lines, and 2 bank address lines) and accessing each address gives 16-bit output. LPDDR memory is mapped to first 64MB of Cortex-M3 data region address starting from 0xA0000000 to 0xA3FFFFFF (64 MB). Read or Write operations to the address locations from 0xA4000000 access the memory from 0xA0000000, because most significant 6 bits of address are ignored. The same repeats for every 64 MB till

0xDFFFFFFF. The memory visible in the rest of the DDR address space (0xA4000000 to 0xDFFFFFFF) is mirrored data of the same LPDDR 64 MB as shown in Figure 7.

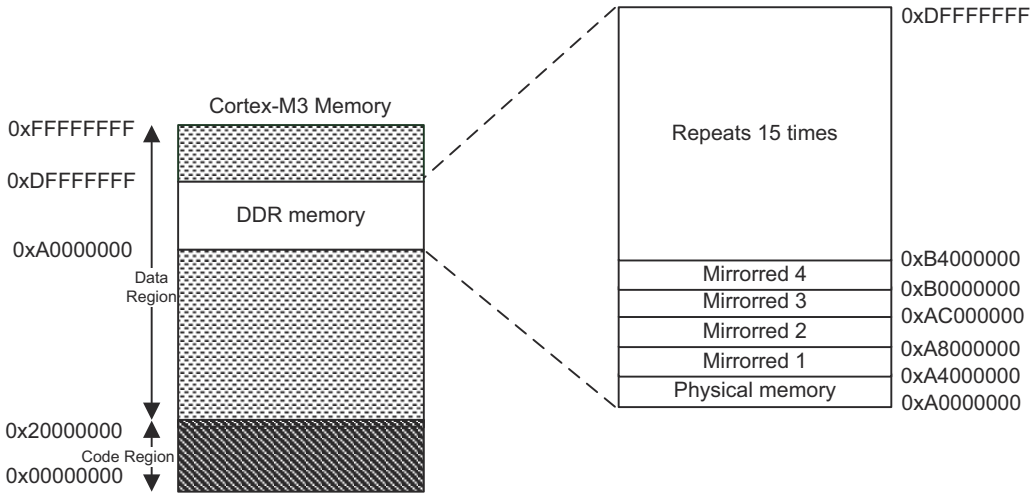

***Figure 7* • Mirroring of DDR**

LPDDR memory (starting at 0xA0000000) is in non-cacheable region (data region) of Cortex-M3. It needs to be remapped to the Cortex-M3 code region to make it cacheable. SmartFusion2 supports remapping of external memory by using the DDR_CR register to the bottom (0x000000000) of the Cortex-M3 code region. Memory mapping of the code region and data regions of Cortex-M3 processor after remapping look as shown in Figure 8.

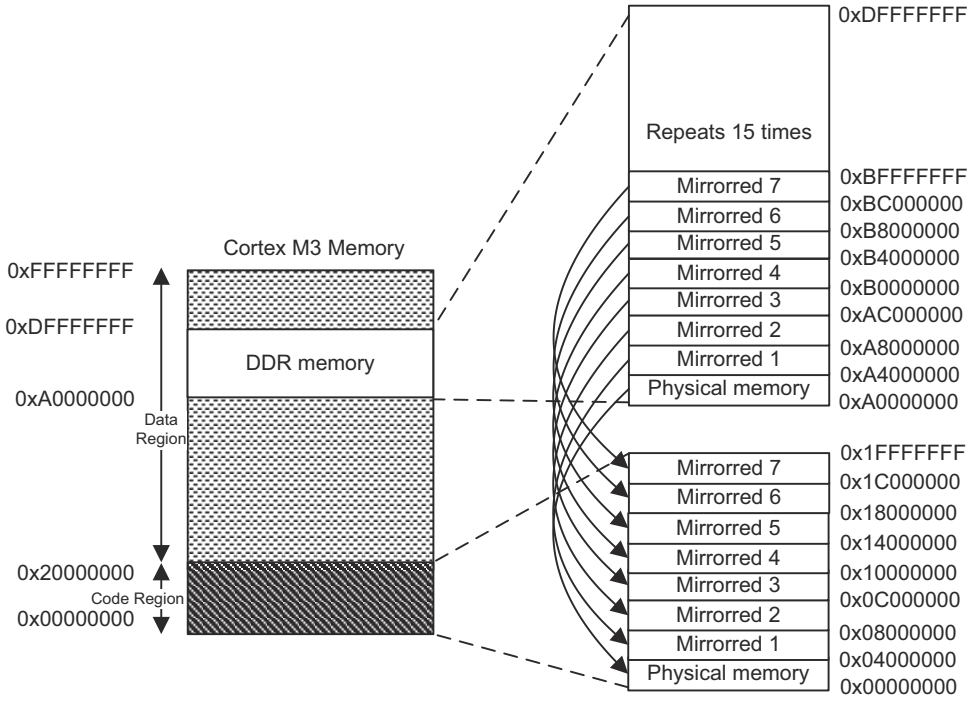

***Figure 8* • Cortex M3 Processor Memory Map After Remapping DDR**

First 512 MB (0xA0000000-0xBFFFFFFF) of DDR memory is remapped to the code region (0x00000000-0x1FFFFFFF) of Cortex-M3 processor as shown in Figure 8.

uclinux image is in the cacheable region of SmartFusion2 after remapping the DDR address space, because it is in the code region of the Cortex-M3 processor. Only code can be cached, anything that is not code (stack, heap, data, and bss) must be accessed from the non-cacheable address region (0xA0000000 to 0xDFFFFFFF). So, the data section is still linked to the data region of Cortex-M3 processor while building the kernel.

Refer to the `linux\arch\arm\mach-m2s\ vmlinux.lds` file for complete details about the linking of data and code sections of the kernel to different memory regions of SmartFusion2.

CONFIG_M2S_CACHE in `linux\arch\arm\mach-m2s\Kconfig` file is used to enable or disable cache.

Conclusion

This application note explains the configuration of Cache Controller and DDR Controller in U-Boot configured for Smartfusion2 Starter Kit.

Appendix A: Design Files

The U-Boot and Linux source files can be downloaded from the Emcraft website:

http://www.emcraft.com/component/jdownloads/view.download/29/484

Design files can be downloaded from the following Emcraft website:

http://www.emcraft.com/jdownloads/som/m2s/m2s-som-2a.zip

The design files consist of Libero Verilog, sample SoftConsole software project, and programming files (*.stp) for SmartFusion2 SoC FPGA Starter Kit.

Prebuilt .stp file with the U-Boot image embedded can be downloaded from the following Emcraft website:

http://www.emcraft.com/jdownloads/som/m2s/m2s-som-2a.stp

Microsemi Corporate Headquarters
One Enterprise, Aliso Viejo CA 92656 USA
Within the USA: +1 (800) 713-4113
Outside the USA: +1 (949) 380-6100
Sales: +1 (949) 380-6136
Fax: +1 (949) 215-4996
E-mail: sales.support@microsemi.com

Microsemi Corporation (Nasdaq: MSCC) offers a comprehensive portfolio of semiconductor and system solutions for communications, defense and security, aerospace, and industrial markets. Products include high-performance and radiation-hardened analog mixed-signal integrated circuits, FPGAs, SoCs, and ASICs; power management products; timing and synchronization devices and precise time solutions, setting the world's standard for time; voice processing devices; RF solutions; discrete components; security technologies and scalable anti-tamper products; Power-over-Ethernet ICs and midspans; as well as custom design capabilities and services. Microsemi is headquartered in Aliso Viejo, Calif. and has approximately 3,400 employees globally. Learn more at **www.microsemi.com**.

51900282-1/03.14