OMAP850

(A Member of Texas Instruments OMAP™ Family of Products)

Technical Reference Manual

Literature Number: SWPU080
Fenruary 2004

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, modifications, enhancements, improvements, and other changes to its products and services at any time and to discontinue any product or service without notice. Customers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All products are sold subject to TI's terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its hardware products to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by government requirements, testing of all parameters of each product is not necessarily performed.

TI assumes no liability for applications assistance or customer product design. Customers are responsible for their products and applications using TI components. To minimize the risks associated with customer products and applications, customers should provide adequate design and operating safeguards.

TI does not warrant or represent that any license, either express or implied, is granted under any TI patent right, copyright, mask work right, or other TI intellectual property right relating to any combination, machine, or process in which TI products or services are used. Information published by TI regarding third-party products or services does not constitute a license from TI to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property of the third party, or a license from TI under the patents or other intellectual property of TI.

Reproduction of information in TI data books or data sheets is permissible only if reproduction is without alteration and is accompanied by all associated warranties, conditions, limitations, and notices. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation.

Resale of TI products or services with statements different from or beyond the parameters stated by TI for that product or service voids all express and any implied warranties for the associated TI product or service and is an unfair and deceptive business practice. TI is not responsible or liable for any such statements.

Following are URLs where you can obtain information on other Texas Instruments products and application solutions:

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DSP	dsp.ti.com	Broadband	www.ti.com/broadband
Interface	interface.ti.com	Digital Control	www.ti.com/digitalcontrol
Logic	logic.ti.com	Military	www.ti.com/military
Power Mgmt	power.ti.com	Optical Networking	www.ti.com/opticalnetwork
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
		Telephony	www.ti.com/telephony
		Video & Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless

Mailing Address: Texas Instruments
Post Office Box 655303 Dallas, Texas 75265

Preface

Read This First

About This Manual

This technical reference manual provides technical information on the OMAP850 multimedia processor.

How to Use This Manual

This document contains the following chapters:

Chapter 1: Introduction to the OMAP850 System

This chapter describes the OMAP850 system architecture, presents the main system features, and discusses system security.

Chapter 2: OMAP3.2.3 and OMAP3.2.4 Differences

This chapter describes the differences between OMAP3.2.3 and OMAP3.2.4.

Chapter 3: Camera Interface

This chapter describes the OMAP850 camera interface.

Chapter 4: LCD Interface

This chapter describes the OMAP850 HR-TFT LCD interface.

Chapter 5: Configuration Registers

This chapter describes the OMAP850 configuration registers.

Chapter 6: MPU-S Memory Mapping

This chapter provides the memory mapping of the OMAP850 MPU subsystem.

Chapter 7: MPU-S Interrupt Mapping

This chapter provides the interrupt mapping of the OMAP850 MPU subsystem.

Chapter 8: MPU-S DMA Requests

This chapter provides the DMA requests of the OMAP850 MPU subsystem.

Chapter 9: GSM-S Memory Mapping

This chapter describes the GSM-S memory mapping of the OMAP850 multimedia processor.

Chapter 10: Frame Buffer

This chapter describes the frame buffer of the OMAP850 multimedia processor.

Appendix A: Pin Descriptions

This appendix presents the OMAP850 platform pin descriptions.

Appendix B: Packaging

This appendix describes the packaging OMAP850 multimedia processor.

Appendix C: OMAP730—OMAP850 Cross-Reference

This appendix contains a copy of the OMAP730TRM (literature number SWPU063B) table of contents (TOC). Shaded text in the appendix indicates those sections of the OMAP730TRM that have been modified to suit the particular features of the OMAP850 devices and that are part of the present document.

Appendix D: OMAP730—OMAP850 Differences

This appendix presents the differences between the OMAP730 and the OMAP850 devices by providing an overview of OMAP850 features for OMAP730 knowledgeable users.

Notational Conventions

This document uses the following conventions.

- Program listings, program examples, and interactive displays are shown in a `special typeface` similar to a typewriter's. Examples use a **`bold version`** of the special typeface for emphasis; interactive displays use a **`bold version`** of the special typeface to distinguish commands that you enter from items that the system displays (such as prompts, command output, error messages, etc.).

  Here is a sample program listing:

```
0011  0005  0001          .field    1, 2
0012  0005  0003          .field    3, 4
0013  0005  0006          .field    6, 3
0014  0006                .even
```

  Here is an example of a system prompt and a command that you might enter:

```
C:  csr -a /user/ti/simuboard/utilities
```

- In syntax descriptions, the instruction, command, or directive is in a **bold typeface** font and parameters are in an *italic typeface*. Portions of a syntax that are in **bold** should be entered as shown; portions of a syntax that are in *italics* describe the type of information that should be entered. Here is an example of a directive syntax:

  .asect **"***section name***"**, *address*

  .asect is the directive. This directive has two parameters, indicated by *section name* and *address*. When you use .asect, the first parameter must be an actual section name, enclosed in double quotes; the second parameter must be an address.

- Square brackets (**[** and **]**) identify an optional parameter. If you use an optional parameter, you specify the information within the brackets; you don't enter the brackets themselves. Here's an example of an instruction that has an optional parameter:

  LALK *16–bit constant [, shift]*

  The LALK instruction has two parameters. The first parameter, *16-bit constant*, is required. The second parameter, *shift*, is optional. As this syntax shows, if you use the optional second parameter, you must precede it with a comma.

  Square brackets are also used as part of the pathname specification for VMS pathnames; in this case, the brackets are actually part of the pathname (they are not optional).

- Braces ({ and }) indicate a list. The symbol **|** (read as *or*) separates items within the list. Here's an example of a list:

```
{ * | *+ | *– }
```

  This provides three choices: `*`, `*+`, or `*–`.

Unless the list is enclosed in square brackets, you must choose one item from the list.

- Some directives can have a varying number of parameters. For example, the .byte directive can have up to 100 parameters. The syntax for this directive is:

  .byte *value$_1$ [, ... , value$_n$]*

  This syntax shows that .byte must have at least one value parameter, but you have the option of supplying additional value parameters, separated by commas.

Information About Cautions and Warnings

This book may contain cautions and warnings.

This is an example of a caution statement.

A caution statement describes a situation that could potentially damage your software or equipment.

This is an example of a warning statement.

A warning statement describes a situation that could potentially cause harm to you.

The information in a caution or a warning is provided for your protection. Please read each caution and warning carefully.

FCC Warning

This equipment is intended for use in a laboratory test environment only. It generates, uses, and can radiate radio frequency energy and has not been tested for compliance with the limits of computing devices pursuant to subpart J of part 15 of FCC rules, which are designed to provide reasonable protection against radio frequency interference. Operation of this equipment in other environments may cause interference with radio communications, in which case the user at his own expense will be required to take whatever measures may be required to correct this interference.

Trademarks

OMAP, TMS320C54x, and C54x are trademarks of Texas Instruments Incorporated.

1-Wire is a registered trademark of Dallas Semiconductor.

All other trademarks are the property of their respective owners.

If You Need Assistance. . .

If you want to. . .	Do this. . .
Request more information about Texas Instruments Digital Signal Processing (DSP) products	Call the CRC† hotline: **(800) 336–5236** Or write to: Texas Instruments Incorporated Market Communications Manager, MS 736 P.O. Box 1443 Houston, Texas 77251–1443
Order Texas Instruments documentation	Call the CRC† hotline: **(800) 336–5236**
Ask questions about product operation or report suspected problems	Call the DSP hotline: **(713) 274–2320**
Report mistakes in this document or any other TI documentation	Fill out and return the reader response card at the end of this book, or send your comments to: Texas Instruments Incorporated Technical Publications Manager, MS 702 P.O. Box 1443 Houston, Texas 77251–1443

† Texas Instruments Customer Response Center

Contents

1 Introduction to the OMAP850 System 1-1
1.1 Detailed Description 1-2
1.1.1 OMAP850 Architecture 1-3
1.2 Features 1-6
1.2.2 Shared Module 1-16
1.3 Architecture 1-17
1.4 Memory Maps 1-18
1.4.1 MPU-S Memory Space 1-18
1.4.2 GSM-S MPU Memory Space 1-18
1.4.3 GSM-S DSP Memory Space 1-19
1.5 EDGE Feature 1-22
1.5.1 Overview 1-22
1.5.2 New OMAP850 Configuration Mode Registers 1-22
1.5.3 DSP Memory-Mapped Registers for GPO 1-23
1.5.4 DAGON Clock and Reset Scheme 1-23
1.5.5 DAGON Functional Interface Handling 1-24
1.5.6 DAGON UART for Trace Handling 1-25

2 OMAP3.2.3/OMAP3.2.4 Differences 2-1
2.1 OMAP3.2.4 Change Summary 2-2
2.2 OMAP3.2.4 Register Changes and Programming Model Differences 2-4
2.2.1 ECN1: 32-Bit Frame Index (DMA) 2-4
2.2.2 ECN2: Packet Hardware Synchronized Transfer (DMA) 2-7
2.2.3 ECN3: Element Counter (DMA) 2-9
2.2.4 ECN4: Removal of Async Mode Logic (CLKRST) 2-10
2.2.5 ECN5: Add Window Tracer Lock Register (WT) 2-10
2.2.6 ECN6: External Clock Gating During Asynchronous Write (EMIFS) 2-11
2.2.7 ECN7: Programmable Ready Sampling Timing (EMIFS) 2-12
2.2.8 ECN8: Synchronous BURST Write in RDMODE 7 (EMIFS) 2-14
2.2.9 ECN9: Abort Handling for Reserved Space (WT) 2-14
2.2.10 ECN11: Control Full Handshaking During Writes (EMIFS) 2-15
2.2.11 ECN13: TC Doubler (EMIFF) 2-17
2.2.12 ECN14: ac Parameter Optimization/SDF133 Support (EMIFF) 2-18
2.2.13 ECN15: Req CAS Extra Cycle Removal 2-21
2.2.14 ECN16: Back to Back Single \Writes From Same Host (EMIFF) 2-22
2.2.15 ECN17: tRP Reduction From 5 to 4 Cycles (EMIFF) 2-22
2.2.16 ECN18: Back to Back Single Reads From Same Host (EMIFF) 2-22
2.2.17 ECN19: TWR Removal for Accesses to Different Banks (EMIFF) 2-23
2.2.18 ECN20: 32-bit FIFO for Each Physical Channel (EMIFF) 2-23

2.2.19 ECN24: External Flash Clock Balancing (EMIFS) 2-24
2.2.20 ECN27: Dynamic Power saving mode (LCD controller) 2-24
2.2.21 ECN28: Adding of Clock Gating Logic Depending on MCMD (OCPI) 2-27
2.2.22 BTS12: 1024 Line Interrupt 2-29
2.2.23 BTS13: OMAP31 Silicon Failure 2-30
2.2.24 OMAP3.2.4 Registers 2-33
2.2.25 Window Tracer Registers (WT) 2-57
2.2.26 EMIFS Registers 2-62
2.2.27 EMIFF Registers 2-71
2.2.28 LCD Controller Registers 2-81
2.2.29 OCPI Registers 2-86

3 Camera Interface 3-1
3.1 Camera Parallel Interface 3-2
3.1.1 Functional Architecture 3-2
3.1.2 Clock Switching Procedures 3-12

4 LCD Interface 4-1
4.1 Introduction 4-2
4.2 Features and Supported Panel Types 4-3
4.2.1 Module Interface 4-4
4.3 Functional Description 4-5
4.4 Registers 4-10
4.5 Programming Model 4-17
4.5.1 OMAP LCD Controller Settings 4-17
4.5.2 Power Management 4-18

5 Configuration Registers 5-1
5.1 OMAP850 Configuration Registers 5-2

6 MPU-S Memory Mapping 6-1
6.1 MPU-S Memory Mapping 6-2
6.1.1 MPU Memory Space 6-2

7 MPU-S Interrupt Mapping 7-1
7.1 MPU-S Interrupt Mapping 7-2

8 MPU-S DMA Requests 8-1
8.1 MPU-S DMA Requests 8-2

9 GSM-S Memory Mapping 9-1
9.1 GSM-MPU Memory Mapping 9-2
9.1.1 GSM-MPU Memory Mapping 9-2
9.2 External Flash ROM Image 9-3
9.3 GSM-S DSP Memory Space 9-7
9.3.1 MPUI Shared Memory 9-9
9.3.2 XIO Memory Mapping 9-9
9.3.3 XIO-TIPB 9-10

10 Frame Buffer 10-1
10.1 Frame Buffer 10-2

A Pin Descriptions A-1
A.1 Pin Descriptions by Module A-2
A.2 Pin Multiplexing A-36

B Packaging B-1
B.1 Package Pin Location B-2
B.2 Mechanical Data B-3

C OMAP730–OMAP850 Cross-Reference C-1
C.1 TOC Cross-Reference C-2

D OMAP730–OMAP850 Differences D-1
D.1 OMAP850/OMAP730 Comparison Overview D-2

Figures

1–1 OMAP850 Overview 1-2
1–2 OMAP850 Wireless-PDA Application Overview 1-3
1–3 OMAP850 Multimedia Processor 1-4
1–4 OMAP850 Intersystem Peripherals 1-5
1–5 MPU-S Memory Map 1-18
1–6 GSM-S Memory Map 1-19
1–7 GPRS-EDGE Feature Additions 1-22
1–8 DAGON Trace 1-24
2–1 Generation of Gated Clock 2-26
2–2 Generation of Non-Gated Clock 2-26
3–1 Image Data Transfer 3-3
3–2 Timing Chart of Image Data Transfer (POLCLK = 1) 3-4
3–3 Order of Camera Data on OCPT1 (Not Swapped) 3-5
3–4 Order of Camera Data on OCPT1 (Swapped) 3-5
3–5 DMA Request 3-6
3–6 FIFO Buffer Parts 3-6
3–7 IRQ Generated on VSYNC Falling Edge 3-7
4–1 OMAP850 HR-TFT Interface High-Level Diagram 4-2
4–2 HR-TFT LCD Interface Functional Diagram 4-5
6–1 MPU Memory Maps 6-3
B–1 OMAP850 Package Pin Location (Bottom View) B-2

Tables

1–1 GSM-S DSP Memory Mapping 1-21
1–2 Data Space 1 1-23
1–3 GPO Register 1-23
1–4 DAGON UART Pin Multiplex 1-25
2–1 OMAP3.2.x and OMAP3.2.4 Change Summary 2-2
2–2 DMA Channel Control Register 2 (DMA_CCR2) 2-5
2–3 Extended Source Frame Index Registers (DMA_CEXSFI_U) 2-5
2–4 Extended Source Frame Index Registers (DMA_CEXSFI_L) 2-6
2–5 Extended Destination Frame Index Registers (DMA_CEXDFI_U (0xFFFE:D83C)) 2-6
2–6 Extended Destination Frame Index Registers (DMA_CEXDFI_L) 2-6
2–7 DMA Channel Control Register 2 (DMA_CCR2) 2-8
2–8 DMA Channel Current Destination Element Number Lower (DMA_CCDEN_L) 2-9
2–9 DMA Channel Current Destination Element Number Upper (DMA_CCDEN_U) 2-9
2–10 WT Window Tracer Lock (WT_LOCK) 2-11
2–11 EMIFS Advanced Chip-Select Configuration FLASH_ACFG_n 2-12
2–12 EMIFS Advanced Chip-Select Configuration FLASH_ACFG_n 2-13
2–13 WT Abort Handling For Reserved Space (RES_SPC_ATYPER) 2-15
2–14 WT Abort Transfer Address (RES_SPC_ADDR) 2-15
2–15 WRRDYMASK Settings 2-16
2–16 EMIFS Dynamic Wait State Register FL_CFG_DYN_WAIT 2-16
2–17 EMIFF Enable and Disable TC Doubler Feature (EMIFF_DOUBLER_EN) 2-17
2–18 EMIFF OMAP3.2 AC Timings (EMIFF_CONFIG_REG2[3] = 0) 2-19
2–19 EMIFF Optimized AC Timings (EMIFF_CONFIG_REG2[3] = 1) 2-20
2–20 Second EMIFF SDRAM Configuration Register (EMIFF_CONFIG_REG2) 2-21
2–21 LCD Control Register (LCD_CTRL_REG) 2-25
2–22 OCPI Dynamic Power Down (DYNAMIC_POWER_DOWN (0xFFFE:C33C)) 2-29
2–23 LCD Display Status Register (DISPLAY_STATUS_REGISTER (0xFFFE:C01C)) 2-29
2–24 LCD Control Register (LCD_CTRL_REG) 2-30
2–25 DMA Controller Registers 2-33
2–26 Channel Source Destination Parameters Register (DMA_CSDP_CHX) 2-35
2–27 Channel Control Register (DMA_CCR_CHX) 2-36
2–28 Channel Interrupt CTRL Register (DMA_CICR_CHX) 2-37
2–29 Channel Status Register (DMA_CSR_CHX) 2-38
2–30 Channel Source Start Address, Lower Bits Register (DMA_CSSA_L_CHX) 2-38
2–31 Channel Source Start Address, Upper Bits Registers (DMA_CSSA_U_CHX) 2-38
2–32 Channel Destination Start Address, Lower Bits Register (DMA_CDSA_L_CHX) 2-39
2–33 Channel Destination Start Address, Upper Bits Register (DMA_CDSA_U_CHX) 2-39
2–34 Channel Element Number Register (DMA_CEN_CHX) 2-39
2–35 Channel Frame Number Register (DMA_CFN_CHX) 2-39

2–36 Channel Frame Index Register (DMA_CFI_CHX) . . . 2-39
2–37 Channel Element Index Register (DMA_CEI_CHX) . . . 2-40
2–38 Channel Progress Counter Register (DMA_CPC_CHX) . . . 2-40
2–39 Channel Destination Address Counter Register (DMA_CDAC_CHX) . . . 2-40
2–40 DMA Channel Destination Element Index Register (DMA_CDEI_CHX) . . . 2-40
2–41 Channel Destination Frame Index Register (DMA_CDFI_CHX) . . . 2-41
2–42 DMA Color Parameter Register Register (DMA_COLOR_L_CHX) . . . 2-41
2–43 DMA Color Parameter Register (DMA_COLOR_U_CHX) . . . 2-42
2–44 DMA Channel Control Register 2 (DMA_CCR2_CHX) . . . 2-42
2–45 Logical Channel Link Control Register (DMA_CLNK_CTRL_CHX) . . . 2-44
2–46 DMA Logical Channel Control Register (DMA_LCH_CTRL_CHX) . . . 2-44
2–47 Channel Current Destination Element Number (Least Significant Word) Register (DMA_CDDEN_L) . . . 2-45
2–48 Channel Current Destination Element Number (Most Significant Word) Register (DMA_CDDEN_U) . . . 2-45
2–49 Extended Source Frame Index (Most Significant Word) Register (DMA_CEXSFI_U) . . . 2-45
2–50 Extended Source Frame Index (Least Significant Word) Register (DMA_CEXSFI_L) . . . 2-45
2–51 Extended Destination Frame Index (Most Significant Word) Register (DMA_CEXDFI_U) . . . 2-46
2–52 Extended Destination Frame Index (Least Significant Word) Register (DMA_CEXDFI_L) . . . 2-46
2–53 LCD Control Register (DMA_LCD_CRTL) . . . 2-46
2–54 LCD Top Address for Frame Buffer 1 Lower Bits Register (DMA_LCD_TOP_F1_L) . . . 2-47
2–55 LCD Top Address for Frame Buffer 1 Upper Bits Register (DMA_LCD_TOP_F1_U) . . . 2-47
2–56 LCD Bottom Address for Frame Buffer 1 Lower Bits Register (DMA_LCD_BOT_F1_L) . . . 2-47
2–57 LCD Bottom Address for Frame Buffer 1 Upper Bits Register (DMA_LCD_BOT_F1_U) . . . 2-48
2–58 LCD Top Address for Frame Buffer 2 Upper Bits Register (DMA_LCD_TOP_F2_L) . . . 2-48
2–59 LCD Top Address for Frame Buffer 2 Upper Bits Register (DMA_LCD_TOP_F2_U) . . . 2-48
2–60 LCD Bottom Address for Frame Buffer 2 Lower Bits Register (DMA_LCD_BOT_F2_L) . . . 2-48
2–61 LCD Bottom Address for Frame Buffer 2 Upper Bits Register (DMA_LCD_BOT_F2_U) . . . 2-48
2–62 Global Control Register (DMA_GCR) . . . 2-49
2–63 Software Compatible Register (DMA_GSCR) . . . 2-49
2–64 Software Reset Control Register (DMA_GRST) . . . 2-50
2–65 DMA Hardware Version ID Number Register (DMA_HW_ID) . . . 2-50
2–66 Pch-P Version ID Register (DMA_PCHP_ID) . . . 2-50
2–67 Pch-M0 Version ID Register (DMA_PCHM0_ID) . . . 2-50
2–68 PCH-M1 Version ID Register (DMA_PCHM_1_ID) . . . 2-50
2–69 PCH-G Version ID Number (DMA_PCHG_ID) . . . 2-50
2–70 DMA Pch-D Version ID Number Register (DMA_PCHID_ID) . . . 2-51
2–71 Global DMA Capability Register 0 (DMA_CAPS_0_U) . . . 2-51
2–72 Global DMA Capability Register 0 (DMA_CAPS_0_L) . . . 2-51
2–73 Global DMA Capability Register 1 (DMA_CAPS_1) . . . 2-52
2–74 Global DMA Capability Register 2 (DMA_CAPS_2) . . . 2-52
2–75 Global DMA Capability Register 3 (DMA_CAPS_3) . . . 2-53
2–76 Global DMA Capability Register 4 (DMA_CAPS_4) . . . 2-54
2–77 DMA Peripheral Physical Channel 0 Status Register (DMA_PCHSR_P_0) . . . 2-55
2–78 DMA Memory Physical Channel 0 Status Register (DMA_PCHSR_M_0) . . . 2-55

2–79 DMA Memory Physical Channel 1 Status Register (DMA_PCHSR_M_1) 2-55
2–80 DMA Physical Display Channel 0 Status Register (DMA_PCHSR_D_0) 2-56
2–81 Window Tracer Registers 2-57
2–82 Top Address of the First Window of EMIFF Register (EMIFF_W1_TOPADDR) 2-57
2–83 Bottom Address of the First Window of EMIFF Register (EMIFF_W1_BOTADDR) 2-57
2–84 Top Address of the Second Window of EMIFF Register (EMIFF_W2_TOPADDR) 2-58
2–85 Bottom Address of the Second Window of EMIFF Register (EMIFF_W2_BOTADDR) 2-58
2–86 Top Address of the First Window of EMIFS Register (EMIFS_W1_TOPADDR) 2-58
2–87 Bottom Address of the First Window of EMIFS Register (EMIFS_W1_BOTADDR) 2-58
2–88 Top Address of the Second Window of EMIFS Register (EMIFS_W2_TOPADDR) 2-58
2–89 Bottom Address of the Second Window of EMIFS Register (EMIFS_W2_BOTADDR) 2-58
2–90 Top Address of the First Window of OCPT1 Register (OCPT1_W1_TOPADDR) 2-58
2–91 Bottom Address of the First Window of OCPT1 Register (OCPT1_W1_BOTADDR) . . . 2-59
2–92 Top Address of the Second Window of OCPT1 Register (OCPT1_W2_TOPADDR) . . . 2-59
2–93 Bottom Address of the Second Window of OCPT1 Register (OCPT1_W2_BOTADDR) 2-59
2–94 Top Address of the First Window of OCPT2 Register (OCPT2_W1_TOPADDR) 2-59
2–95 Bottom Address of the First Window of OCPT2 Register (OCPT2_W1_BOTADDR) . . . 2-59
2–96 Top Address of the Second Window of OCPT2 Register (OCPT2_W2_TOPADDR) . . . 2-59
2–97 Bottom Address of the Second Window of OCPT2 Register (OCPT2_W2_BOTADDR) 2-59
2–98 Window Enable Register of WT (WT_WINENR) 2-60
2–99 Window Tracer Lock Register (WT_LOCK) 2-60
2–100 Abort Handling for Reserved Space Register (RES_SPC_ATYPER) 2-60
2–101 Reserved Space Address Register (RES_SPC_ADDR) 2-61
2–102 EMIFS Registers 2-62
2–103 EMIFS LRU Priority Register (EMIFS_LRUREG) 2-62
2–104 EMIFS Configuration Register (EMIFS_CONFIG) 2-63
2–105 EMIFS Chip-Select Configuration Register nCS0 (FLASH_CFG_0) 2-63
2–106 EMIFS Chip-Select Configuration Register nCS1 (FLASH_CFG_1) 2-64
2–107 EMIFS Chip-Select Configuration Register nCS2 (FLASH_CFG_2) 2-65
2–108 EMIFS Chip-Select Configuration Register nCS3 (FLASH_CFG_3) 2-66
2–109 EMIFS Dynamic Priority time-out 1 (EMIFS_TIMEOUT1_REG) 2-67
2–110 EMIFS Dynamic Priority time-out 2 (EMIFS_TIMEOUT2_REG) 2-67
2–111 EMIFS Dynamic Priority time-out 3 (EMIFS_TIMEOUT3_REG) 2-67
2–112 EMIFS Dynamic Wait State Register (FL_CFG_DYN_WAIT) 2-68
2–113 EMIFS Abort Address Register (EMIFS_ABORT_ADDR) 2-68
2–114 EMIFS Abort Type Register (EMIFS_ABORT_TYPE) 2-68
2–115 EMIFS Abort time-out Register (EMIFS_ABORT_TOUT) 2-68
2–116 EMIFS Advanced Chip Configuration Register nCS0 (FLASH_ACFG_0_I) 2-69
2–117 EMIFS Advanced Chip Configuration Register nCS1 (FLASH_ACFG_1_I) 2-69
2–118 EMIFS Advanced Chip Configuration Register nCS2 (FLASH_ACFG_2_I) 2-70
2–119 EMIFS Advanced Chip Configuration Register nCS3 (FLASH_ACFG_3_I) 2-70
2–120 EMIFF Registers 2-71
2–121 EMIFF Priority Register (EMIFF_PRIORITY_REG) 2-72
2–122 EMIFF SDRAM Configuration Register (EMIFF_SDRAM_CONFIG) 2-72
2–123 EMIFF SDRAM MRS Register (EMIFF_MRS) 2-73
2–124 EMIFF SDRAM Configuration Register (EMIFF_CONFIG_REG2) 2-73
2–125 EMIFF TC Doubler Register (EMIFF_DOUBLER_EN) 2-73

2–126 DLL Write Operations Register (DLL_WRT_CONTROL) 2-74
2–127 DLL Write Status Register (DLL_WRT_STATUS) 2-74
2–128 EMIF Fast Interface SDRAM MRS Register (EMIFF_MRS_NEW) 2-75
2–129 Extended Mode Register 0 (EMIFF_EMRS0) 2-75
2–130 Extended Mode Register 1 (EMIFF_EMRS1) 2-76
2–131 SDRAM Operation Register (SDRAM_OPERATION_REG) 2-76
2–132 EMIFF Manual Command Register (SDRAM_MANUAL_CMD_REG) 2-77
2–133 EMIFF TIMEOUT Register 1 (EMIFF_TIMEOUT1) 2-77
2–134 EMIFF TIMEOUT Register 2 (EMIFF_TIMEOUT2) 2-77
2–135 EMIFF TIMEOUT Register 3 (EMIFF_TIMEOUT3) 2-77
2–136 EMIFF Aborted Transfer Address Register (EMIFF_ABORT_ADDRESS) 2-78
2–137 EMIFF Aborted Type Register (EMIFF_ABORT_TYPE) 2-78
2–138 DLL Lower Byte Status Register (DLL_LRD_STATUS) 2-78
2–139 DLL Upper Byte Control Register (DLL_URD_CONTROL) 2-78
2–140 DLL Upper Byte Status Register (DLL_URD_STATUS) 2-79
2–141 EMIFF Reserved Extended Mode Register (EMIFF_EMRS2) 2-79
2–142 DLL Lower Byte Control Register (DLL_LRD_CONTROL) 2-80
2–143 LCD Controller Registers 2-81
2–144 LCD Control Register (LCD_CTRL_REG) 2-81
2–145 LCD Timing 0 Register (LCD_TIMING_0) 2-83
2–146 LCD Timing 1 Register (LCD_TIMING_1) 2-83
2–147 LCD Timing 2 Register (LCD_TIMING_2) 2-83
2–148 LCD Status Register (LCD_STATUS) 2-84
2–149 LCD Sub-Panel Display Register (LCD_SUB_PANEL) 2-85
2–150 Line Interrupt Register (LINE_INTERRUPT_REGISTER) 2-85
2–151 Display Status register (DISPLAY_STATUS_REGISTER) 2-85
2–152 OCPI Registers 2-86
2–153 Address Fault Register (ADDRFAULT) 2-86
2–154 Master Command Fault Register (MCMDFAULT) 2-86
2–155 Interrupt Configuration 0 Register (SINTERRUPT0) 2-87
2–156 Type of Abort Register (ABORTTYPE) 2-87
2–157 Interrupt Configuration 1 Register (SINTERRUPT1) 2-88
2–158 Memory Space Protection Register (PROTECT) 2-88
2–159 Secure Mode Control Register (SECURE_MODE) 2-88
2–160 Dynamic Power Down Register (DYNAMIC_POWER_DOWN) 2-89
3–1 Default Configuration at Reset 3-8
3–2 Camera Interface Registers 3-9
3–3 Clock Control Register (CTRLCLOCK) 3-9
3–4 Interrupt Source Status Register (IT_STATUS) 3-10
3–5 Camera Interface Mode Configuration Register (MODE) 3-10
3–6 Status Register (STATUS) 3-11
3–7 Camera Interface GPIO Register (GPIO) 3-11
3–8 Image Data Register (CAMDATA) 3-11
3–9 FIFO Peak Counter Register (PEAK_COUNTER) 3-12
4–1 Supported Panel Types 4-3
4–2 HR-TFT LCD Interface Signals 4-4
4–3 LCD Driver Sub-Block Mapping (Bypass Mode) 4-6
4–4 HR-TFT Horizontal Scanning Direction Selection 4-7
4–5 LCD Driver Sub-Block Output Mapping of Control Signals (HR-TFT Mode) 4-8
4–6 LCD Driver Control and Timing Registers 4-10

4–7 LCD Driver Functional Control Register 0 (LCDDRV_FNCTL0) 4-10
4–8 LCD Driver Functional Control Register 1 (LCDDRV_FNCTL1) 4-13
4–9 LCD Driver Timing_Lp Control Register (LCDDRV_TIMLP) 4-13
4–10 LCD Driver Timing_Cls Control Register (LCDDRV_TIMCLS) 4-14
4–11 LCD Driver Timing_Ps Control Register (LCDDRV_TIMPS) 4-14
4–12 LCD Driver Timing_Spir Control Register (LCDDRV_TIMSPLR) 4-14
4–13 LCD Driver Timing_Rev Control Register (LCDDRV_TIMREV) 4-14
4–14 LCD Driver Timing_Sps Control Register (LCDDRV_TIMSPS) 4-15
4–15 LCD Driver Timing_PS2PS3 Control Register (LCDDRV_TIMPS2PS3) 4-15
4–16 LCD Driver Timing_PS2 Control Register (LCDDRV_TIMPS2) 4-15
4–17 LCD Driver Timing_ASC1 Control Register (LCDDRV_TIMASC1) 4-15
4–18 LCD Driver Timing_ASC2 Control Register (LCDDRV_TIMASC2) 4-16
4–19 LCD Driver Timing_SSC1 Control Register (LCDDRV_TIMSSC1) 4-16
4–20 LCD Driver Timing_SSC2 Control Register (LCDDRV_TIMSSC2) 4-16
4–21 OMAP850 LCD Control Register (LCDCONTROL) 4-17
4–22 OMAP850 LCD Timing 2 Register (LCDTIMING2) 4-17
5–1 OMAP850 Configuration Registers 5-2
5–2 Device Identification on MPU–S side Register (PERSEUS2_MPU_DEV_ID) 5-4
5–3 Device Identification 0 (LSBs) on GSM-S side Register (PERSEUS2_GSM_DEV_ID0) 5-4
5–4 Device Identification 1 (LSBs) on GSM–S side Register (PERSEUS2_GSM_DEV_ID1) 5-4
5–5 Software Compatibility with EDGE Register (DSP_CONF) 5-4
5–6 OMAP850 Die Identification 0 on MPU–S Side Register (PERSEUS2_MPU_DIE_ID0) . 5-4
5–7 Software Compatibility with TBB2100 Register (GSM_ASIC_CONF) 5-5
5–8 OMAP850 Die Identification 1 on MPU–S Side Register (PERSEUS2_MPU_DIE_ID1) 5-5
5–9 OMAP850 Mode Configuration 1 Register (PERSEUS2_MODE1) 5-5
5–10 OMAP850 Die Identification 0 on GSM-S Side Register (PERSEUS2_GSM_DIE_ID0) 5-7
5–11 OMAP850 Die Identification 1 on GSM-S Side Register (PERSEUS2_GSM_DIE_ID1) 5-7
5–12 OMAP850 Mode Configuration 2 Register (PERSEUS25_MODE2) 5-7
5–13 OMAP850 Die Identification 2 on GSM-S Side Register (PERSEUS2_GSM_DIE_ID2) 5-9
5–14 OMAP850 Die Identification 3 on GSM-S side Register (PERSEUS2_GSM_DIE_ID3) 5-9
5–15 Analog Cells Configuration Register (PERSEUS2_ANALOG_CELLS_CONF) 5-9
5–16 Secure Control Register (SECCTRL) 5-10
5–17 Spare 1 Register for ECO (SPARE1) 5-14
5–18 Spare 2 Register for ECO (SPARE2) 5-15
5–19 Edge Register (GSM_PBG_IRQ) 5-16
5–20 DMA Mode Configuration Register (DMA_REQ_CONF) 5-16
5–21 Pull Enable Control Register (PE_CONF_NO_DUAL) 5-16
5–22 OMAP850 Shared I/O Configuration 0 Register (PERSEUS2_IO_CONF0) 5-17
5–23 OMAP850 Shared I/O Configuration 1 Register (PERSEUS2_IO_CONF1) 5-19
5–24 OMAP850 Shared I/O Configuration 2 Register (PERSEUS2_IO_CONF2) 5-21
5–25 OMAP850 Shared I/O Configuration 3 Register (PERSEUS2_IO_CONF3) 5-23
5–26 OMAP850 Shared I/O Configuration 4 Register (PERSEUS2_IO_CONF4) 5-24
5–27 OMAP850 Shared I/O Configuration 5 Register (PERSEUS2_IO_CONF5) 5-27
5–28 OMAP850 Shared I/O Configuration 6 Register (PERSEUS2_IO_CONF6) 5-28
5–29 OMAP850 Shared I/O Configuration7 Register (PERSEUS2_IO_CONF7) 5-30

5–30 OMAP850 Shared I/O Configuration 8 Register (PERSEUS2_IO_CONF8) 5-32
5–31 OMAP850 Shared I/O Configuration 9 Register (PERSEUS2_IO_CONF9) 5-34
5–32 OMAP850 Shared I/O Configuration 10 Register (PERSEUS2_IO_CONF10) 5-35
5–33 OMAP850 Shared I/O Configuration 11 Register (PERSEUS2_IO_CONF11) 5-37
5–34 OMAP850 Shared I/O Configuration 12 Register (PERSEUS2_IO_CONF12) 5-39
5–35 OMAP850 Shared I/O Configuration 13 Register (PERSEUS2_IO_CONF13) 5-41
5–36 48-MHz Input Control Register (PERSEUS_PCC_CONF_REG) 5-42
5–37 BIST Fail Go Register (BIST_STATUS_INTERNAL) 5-43
5–38 BIST Settings Control Register (BIST_CONTROL) 5-44
5–39 Boot Procedure Register (BOOT_ROM_REG) 5-45
5–40 Secure chip (PRODUCTION_ID_REG) 5-46
5–41 Secure ROM Signature 1 Register (BIST_SECROM_SIGNATURE1_INTERNAL) 5-47
5–42 Secure ROM Signature 2 Register (BIST_SECROM_SIGNATURE2_INTERNAL) 5-47
5–43 BIST Settings Control Register (BIST_CONTROL_2) 5-47
5–44 Debug Signal Selection 1 Register (DEBUG1) 5-48
5–45 Debug Signal Selection 2 Register (DEBUG2) 5-49
5–46 DMA and IRQ Selection Register (DEBUG_DMA_IRQ) 5-49
5–47 Software Compatibility Purpose Register (PERSEUS25_MODE) 5-50
5–48 8 to 1 Address Configuration Register(PERSEUS25_FADD_IOCONF1) 5-52
5–49 15 to 9 Address Configuration Register (PERSEUS25_FADD_IOCONF2) 5-53
5–50 TWC310 Control Register (PERSEUS25_DAGON_MODE) 5-55
5–51 OMAP850 Shared I/O Configuration 0 Register (PERSEUS25_DAGON_IO_CONF0) 5-55
5–52 OMAP850 Shared I/O Configuration 1 Register (PERSEUS2_DAGON_IO_CONF1) 5-56
5–53 TWC310 Dual Debug Signal Selection 0 Register (DEBUG_DAGON_0) 5-57
5–54 TWC310 Dual Debug Signal Selection 1 Register (DEBUG_DAGON_1) 5-58
5–55 OMAP850 Shared I/O Configuration Register (PCONF_DAGON_JTAG_CTRL) 5-59
6–1 Memory Bus Associated With Chip-Selects 6-4
6–2 MPU Memory Address Space 6-4
6–3 TIPB Peripherals Address Space 6-5
7–1 MPU-S Incoming Interrupts 7-2
8–1 Activated External DMA Requests 8-2
8–2 MPU-S DMA Requests 8-2
9–1 GSM-MPU Memory Map 9-2
9–2 Data Format 9-3
9–3 GSM-MPU Peripheral Mapping (Strobe 0) 9-4
9–4 GSM-MPU Peripheral Mapping (Strobe 1) 9-5
9–5 DSP Memory Mapping 9-7
9–6 DSP XIO Memory Space 9-10
A–1 Pin Descriptions by Module A-2
A–2 TPU/TSP A-3
A–3 GSM–S Voice AuSPI A-5
A–4 GSM–S RIF BB A-5
A–5 GSM-S SIM A-5
A–6 GSM-S MCSI A-6
A–7 GSM-S UART A-6
A–8 GSM-S μWIRE A-7
A–9 GSM-S LPG A-7
A–10 GSM-S I^2C A-7
A–11 GSM-S GPIO A-8

A–12 MPU-S LCD A-9
A–13 MPU-S HR TFT LCD A-9
A–14 MPU-S UART Modem-IrDA A-11
A–15 MPU-S UART Modem A-11
A–16 MPU-S SPI 100K 1 . . . . . A-12
A–17 MPU-S SPI 100K 2 . . . . . A-12
A–18 MPU-S MMC/SDIO A-13
A–19 MPU-S I^2C A-13
A–20 MPU-S HDQ-1WIRE A-13
A–21 MPU-S μWIRE A-14
A–22 MPU-S PWL A-14
A–23 MPU-S PWT A-14
A–24 MPU-S LPG A-14
A–25 MPU-S Extended GPIO A-15
A–26 MPU-S EAC BT AuSPI A-15
A–27 MPU-S EAC Audio Codec A-16
A–28 MPU-S USB OTG A-16
A–29 MPU-S McBSP1 . . . . . A-18
A–30 MPU-S McBSP2 . . . . . A-18
A–31 MPU-S MCSI A-19
A–32 MPU-S SDRAM A-19
A–33 MPU-S DDR A-22
A–34 MPU-S EMIFS A-22
A–35 MPU-S NAND Flash A-26
A–36 MPU-S Camera A-27
A–37 MPU-S MPUIO A-28
A–38 MPU-S Keypad A-29
A–39 MPU-S SMC A-30
A–40 MPU-S ETM9 . . . . . A-30
A–41 MPU-S VLYNQ A-31
A–42 MPU-S SSPI A-31
A–43 MPU-S TAP A-32
A–44 Miscellaneous A-32
A–45 Powers A-34
A–46 OMAP780 Pin Multiplexing A-36

Chapter 1

Introduction to the OMAP850 System

This chapter describes the OMAP850 multimedia processor system architecture, presents the main system features, and discusses system security.

Topic **Page**

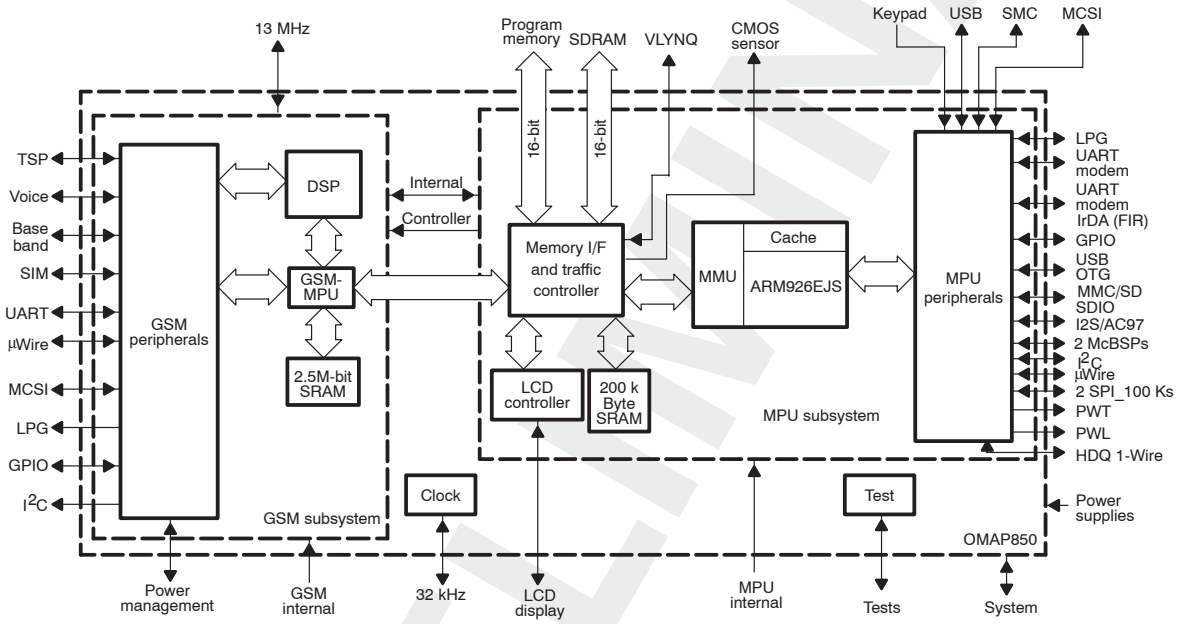
1.1	Detailed Description	1-2
1.2	Features	1-6
1.3	Architecture	1-17
1.4	Memory Maps	1-18
1.5	EDGE Feature	1-22

1.1 Detailed Description

The OMAP850 is the Texas Instruments solution for wireless pocket information devices such as wireless PDA, smart phones, Java-enabled web phones and other wireless handsets that combine both voice and data. The OMAP850 is designed to run Windows CE, EPOC, Palm, Linux, and other operating systems.

The OMAP850 consists of two main subsystems that share external memories with the help of a memory and traffic controller. The GSM subsystem (GSM-S) handles the complete GSM protocol stack and signal processing. The MPU subsystem (MPU-S) runs the OS-controlling application tasks and all non-GSM peripherals.

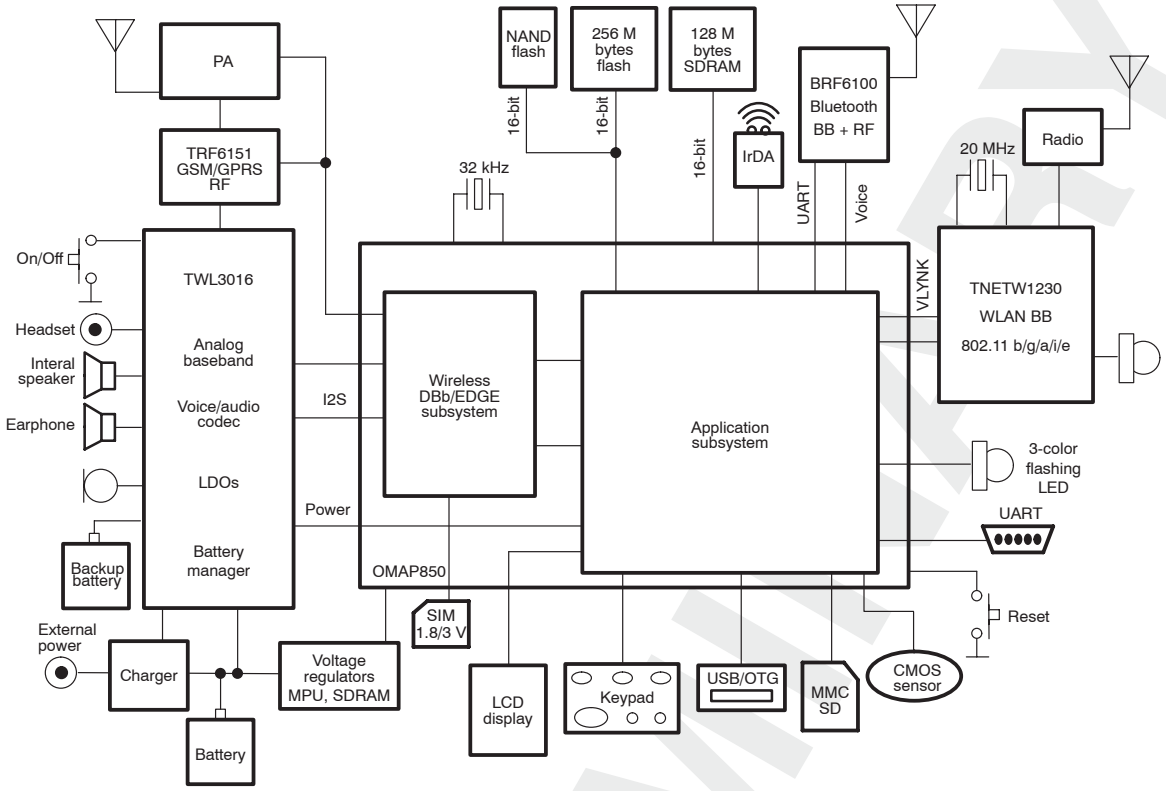
Figure 1–1. OMAP850 Overview



GSM-S external peripherals include the RF module, the TWL3016 or ABB analog baseband/audio A/D-D/A and battery manager chip, the touch screen, a serial link, and a SIM card interface. The MPU-S controls peripherals such as display, keyboard/keypad, IrDA or other serial links. Other peripherals such as memory card or GPS can also be connected to the system.

OMAP850 contains a set of secure modules, including ROM, a single port SRAM, and eFUSE cells. These components enable the system to support secure applications.

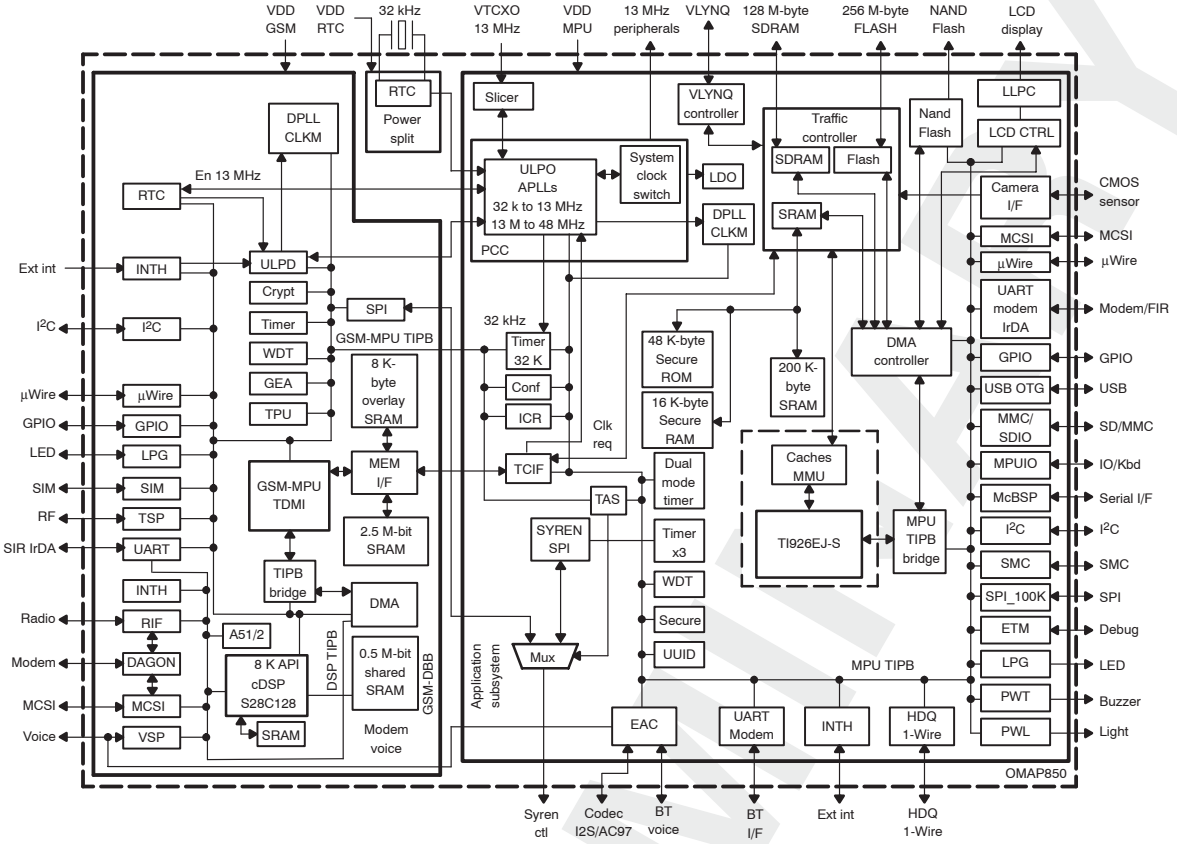
Figure 1–2. OMAP850 Wireless-PDA Application Overview



1.1.1 OMAP850 Architecture

OMAP850 is based on the OMAP 3.2.4 MPU core and the TBB2100 GSM baseband subsystem (GSM-MPU + TMS320C54x DSP).

Figure 1–3. OMAP850 Multimedia Processor



1.1.1.1 GSM Subsystem

The GSM subsystem implements the digital baseband processes of a GSM/ GPRS mobile phone. This subsystem combines a DSP subchip (TMS320C54x DSP CPU) with its program and data memories, a microcontroller core with emulation facilities (GSM-MPU TDMIE), internal 8K bytes of overlay boot SRAM memory, up to 2.5M bits of SRAM memory, 0.5M bits of SRAM memory sharable between DSP and GSM-MPU, a clock squarer cell, and several compiled single-port or 2-port RAM and CMOS gates.

This subsystem is used in the management of the GSM/GPRS baseband processes through the GSM layer 1, 2, and 3 protocols, as described in the ETSI standard with specific attention to power consumption in both GSM dedicated and idle modes, and GPRS (class 12) capability.

The GSM subsystem has capability for enhanced GPRS/EDGE protocol processing capability.

The GSM subsystem fully supports the GSM full-level test approval (FTA) for full-rate (FR), enhanced full-rate (EFR) and half-rate (HR) speech coding. It implements all features for structural test of the logic (full-scan, BIST, PMT, JTAG boundary-scan).

1.1.1.2 MPU Subsystem

The MPU-S performs all personal communication system tasks such as call manager, email/fax reader/composer, Internet access, personal digital

assistant (PDA) or personal information management (PIM). The MPU-S also controls the GSM subsystem.

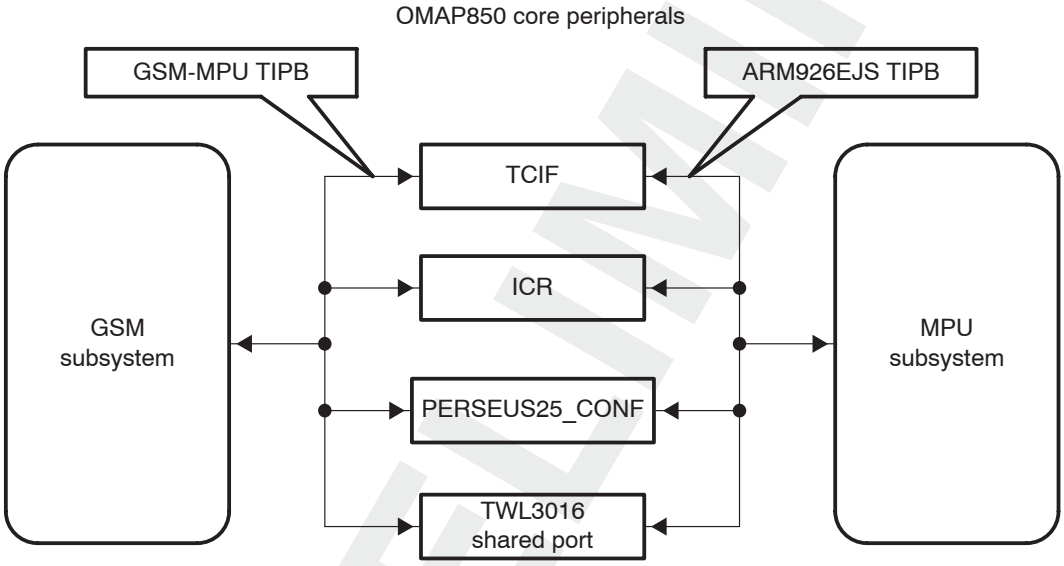
The MPU-S comprises an OMAP 3.2.4 MPU subsystem plus some MPU peripherals.

1.1.1.3 OMAP850 Intersystem Peripherals

In addition to the GSM and the MPU subsystems, the following modules are used for GSM-S/MPU-S interworking management:

- Traffic controller interface (TCIF)
- Intersystem control register (ICR)
- Configuration registers (PERSEUS25_CONF)
- TWL3016 shared port (SSPI)

Figure 1–4. OMAP850 Intersystem Peripherals



1.1.1.4 OMAP850 Processor Peripherals

- JTAG interface module (JTAG)
- Test and debug blocs
- e-Fuse modules
- Boot ROM and security

1.2 Features

1.2.1 GSM MPU Module

The following GSM-MPU modules are used on GSM applications:

- ❑ GSM-MPU TGMIE megamodule
  - ■ GSM-MPU TDMI CPU core (32-/16-bit RISC processor)
  - ■ GSM-MPU ICECrusher for emulation purposes
- ❑ GSM-MPU memory interface for embedded SRAM and external MPU subsystem shared flash and SDRAM access management (through TCIF).
- ❑ 2.5M-bit static RAM with write buffer
- ❑ 0.5M-bit static RAM sharable with DSP with mutual exclusive access
- ❑ 64K-bit static RAM with external memory overlay for internal boot
- ❑ Memory protection unit (MPU)
- ❑ Debug unit (DU)
- ❑ Die-ID cell read access
- ❑ TIPB bridge
- ❑ DMA controller (4 channels, 2 ports)
- ❑ GSM-MPU interrupt handler (INTH)
- ❑ Watchdog timer (WDT)
- ❑ Two generic timers
- ❑ GSM real-time sequencer (TPU)
- ❑ GSM real-time serial port (TSP)
- ❑ SIM interface (SIM)
- ❑ GSM-MPU serial port interface (SPI)
- ❑ UART 16C750 modem with hardware flow control (DCD, CTS/RTS) and autobaud capability shared with DSP
- ❑ Real-time clock (RTC)
- ❑ GSM ultralow-power-down controller (ULPD)
- ❑ Clock generator and control with DPLL (CLKM)
- ❑ Programmable controller for three-color LED pulse generation (LPG)
- ❑ Master I2C serial interface
- ❑ Microwire interface
- ❑ GPRS encryption coprocessor (GEA 1&2)

1.2.1.1 DSP Subchip

The GSM subsystem has the following ASIC DSP features:

- TMS320C54x DSP module (S28C128)
  - TMS320C54x DSP core
  - 28K words of embedded RAM
  - 128K works of ROM
  - 0.5M-bit static RAM sharable with MPU with mutual exclusive access
  - MPUI (8K words, part of the 28K-word RAM)
  - SPI
  - Timer
- TIPB bridge
- Radio interface (RIF)
- Multichannel serial interface (MCSI)
- A51/A52 ciphering (CRYPT)
- UART 16C750 modem with hardware flow control (DCD, CTS/RTS) and autobaud capability shared with MPU
- DMA controller (four channels)
- TMS320C54x DSP interrupt handler (INTH)
- Enhanced GPRS/EDGE protocol capability (DAGON)

1.2.1.2 MPU Module

The MPU-S comprises an OMAP3.2 MPU subsystem plus MPU peripherals.

The OMAP3.2 subsystem contains:

- ARM326EJ MPU:
  - ARM926EJ megacell including:
    - ARM926EJS core, running at 201.5 MHz maximum frequency
    - MMU with translation lookaside buffer (TLBx)
    - L1 16K-byte, four-way set-associative instruction cache
    - L1 8K-byte, four-way set-associative data cache with write buffer
  - MPU level 1 interrupt handler
  - Coprocessor15 (CP15) and protection module
  - 17-word write buffer (WB)
  - System bus interface

- Memory and traffic controller (TC)
  - 16-bit data width memory interface for 128M bytes of addressable SDRAM
  - 16-bit data width memory interface for 256M bytes of addressable flash/RAM/ROM
- Color LCD controller: 2/4/8/16 and pseudo 18 bits/pixel
- MPU interrupt handler: 32 lines (INTH)
- MPU TIPB bridge: 32-bit
- Three 32-bit timers
- Watchdog timer
- Clock generator with DPLLs and power management
- 200K bytes of SRAM with frame buffer DMA channel
- System DMA controller
- Embedded trace macrocell module, ETM version 2.a in 13-bit mode configuration or in 17-bit demultiplexed mode configuration

The MPU-S subsystem contains:

- OMAP3.2.4 data-processing core
- Power and clock control (PCC) with the following main functions:
  - Performs the transitions between the power modes (awak, big sleep, deep sleep)
  - Handles idle/wake-up handshake of MPU-S and DBB
  - Monitors wake-up events
  - Controls system clock input sources (VCTXO, 32-kHz oscillator)
  - Performs calibration of 32-kHz oscillator (gauging)
  - Manages the clocks and resets distributed to MPU-S, DBB, and to some peripherals
  - Manages security resets and violations
  - Handles power-up sequence
  - Manages the 32-kHz oscillator-to-VCTXO, and VCTXO-to-32-kHz oscillator clock switch
  - Handles embedded LDO with bypass possibility
  - Implements full PMT wrapper for all analog cells embedded inside it
- Two UART modems – SIR/MIR/FIR IrDA with the following main features:
  - Selectable UART/IrDA modes.
  - Dual 64-entry FIFOs for received and transmitted data payload

- Programmable and selectable transmit and receive FIFO trigger levels for DMA and interrupt generation
- Programmable sleep mode
- Complete status-reporting capabilities in both normal and sleep modes
- Frequency prescaler values from 0 to 16383 to generate the appropriate baud rates
- Single 48-MHz clock reference for baud setting
- Two DMA requests and one interrupt request to system
- UART/Modem functions:
  - Baud rate from 300 bits/s up to 3.6864 Mbits/s
  - Autobaud between 1200 bits/s and 115.2 Kbits/s
  - Software/Hardware flow control:

Programmable XON/XOFF characters

Programmable auto-$\overline{\text{RTS}}$ and auto-$\overline{\text{CTS}}$

  - Programmable serial interface characteristics:
    - 5-, 6-, 7- or 8-bit characters
    - Even, odd, or no parity bit generation and detection
    - 1-, 1.5-, or 2-stop bit generation
  - False-start bit detection
  - Line-break generation and detection
  - Fully-prioritized interrupt system controls
  - Internal test and loopback capabilities
  - Modem control functions ($\overline{\text{CTS}}$, $\overline{\text{RTS}}$, $\overline{\text{DSR}}$, $\overline{\text{DTR}}$, $\overline{\text{RI}}$ and $\overline{\text{DCD}}$)
- IrDA functions:
  - Slow infrared (SIR: 115.2K baud), medium infrared (MIR: 0.576M baud) and fast infrared (FIR: 4M baud) operations (very fast infrared (VFIR) not supported)
  - Pulse width either 1.6 us or 3/16th of a single bit time
  - Framing error, cyclic redundancy check (CRC) error, illegal symbol (FIR), abort pattern (SIR, MIR) detection
  - Eight-entry status FIFO (with selectable trigger levels) available to monitor frame length and frame errors

❑ General-purpose I/Os (GPIO) with interrupt support: Maskable interrupt generation on high-to-low or low-to-high transition of pins configured as input

The peripherals GPIO modules have ARMPER_CK as their input clock. These GPIOs are asynchronous, and the modules do not perform input line debouncing. These GPIOs, then, can wake up the system on an input level change when ARMPER_CK is off.

- USB OTG controller with 48-MHz APLL. The On-The-Go (OTG) supplement allows a USB peripheral to have the following enhancements:
  - Limited host capability to communicate with other selected USB peripherals
  - A small USB connector to fit the mobile form factor
  - Low power features to preserve battery life
- Multichannel buffered serial port (McBSP)
  - Based on multichannel buffered serial port TI standard
  - Simultaneous RX and TX DMA support
  - Each processor master of its TX clock and Tx data
  - Supports bit rates up to 5M bits/sec
  - RX data overrun interrupt
  - Functionally identical to C5510 McBSP
- Enhanced audio controller (EAC): Supports I2S, AC97, and SPDIF codecs. The EAC provides an application with stand-alone audio control without any CPU (DSP or MCU) support. It allows connecting a modem and/or Bluetooth subsystem to an AC97, a PCM, or an I2S codec, while the MPU subsystem (PDA, WinCE MCU, etc…) is in power-down mode. It also suppresses the need for two codecs (one for the modem and one for the MPU subsystem).

  The EAC also provides the ability to record/play PCM (wave) files with various sample frequencies and bit length formats without any CPU processing. It also allows using the same microphone input line for the modem 8-kHz sampling frequency or for the high quality voice recording.

  A high-quality audio file also can be mixed with the voice input-channel, down-sampled and sent to the modem/Bluetooth uplink path (the wave file plays during a phone call). The EAC also provides the CPU the ability to get the data samples coming from the microphone path at the codec sampling frequency, to process the signal and to send it back to the modem uplink path. The audio signal coming from the modem/Bluetooth downlink path can also be processed by the CPU at 8 kHz or at the codec sampling frequency, before being sent to the codec and loudspeakers.

  Modem/Bluetooth inputs and/or output voice data samples can be automatically stored in the CPU memory through the DMA write channel (record the phone call).

  The EAC also provides the ability to loop back the Bluetooth audio data samples to the modem uplink path.

- Multimedia memory card and secure digital I/O host controller (MMC/SD & 4-bit SDIO) supports the following combination of external devices:
  - One or more MMC memory cards sharing the same bus plus up to four devices with 8-bit SPI protocol interface (serial flash memories, etc)
  - One single SD memory card or SDIO card plus up to four devices with 8-bit SPI protocol interface

  The main features of the multimedia memory card and secure digital I/O host controller are:
  - Full compliance with MMC command/response sets as defined in *Multimedia Card-System Specification, MMCA Technical Committee Version 3.1, June 2001*
  - Full compliance with SD command/response sets as defined in *SD Mmeory Card Specification-Part 1, Physical Layer Specification, SD group, Version 1.0, March 2000*, and *Supplementary Notes-Part 1, Physical Layer Specification, SD Group, June 2000*
  - Full compliance with SDIO command/response sets and interrupt/ read-wait mode as defined in *SDIO Card Specification Part E1, SDIO Working Group, Version 1.0, October 2001*
  - Flexible architecture that allows support for new command structure
  - Separate SPI interface with four CS. Provides supports for up to four serial flash devices.
  - Built-in 64-byte FIFO for buffered read or write
  - 16-bit wide access bus to maximize bus throughput
  - Low-power design
  - Wide-interrupt capability
  - Programmable clock generation
  - Two DMA channels
  - Big-/little-endian mode for data

  Known limitations:
  - No built-in hardware support for error correction codes (ECC)
- NAND flash controller: The NAND-type flash memory chip contributes to the rapid write and erase capabilities since data is rewritten in small increments, leading to improved performance through continuous data recording and other benefits.
- Fast I^2C master/slave controller: External components attached to the I^2C bus can serially transmit/receive up to 8-bit data to/from the local host (LH) device through the two-wire I^2C interface (400K bits/s). This I^2C peripheral supports any slave or master I^2C compatible device.
- SMC (SmartCard) Controller with direct I/O interface: The key to secure network access, it provides secure authentication of the user to the net-

work, enabling the delivery of personalized services, as well as providing data storage for address books. Extending this concept, the SmartCard (SMC) application connects any 3G terminal with any 3G service.

- μWire controller: This serial synchronous interface can drive four serial external components (EEPROM or LCD with uWIRE standard). For the external devices, this interface is compatible with the μWire standard and is seen as the master.
  - Two chip-selects: Each CS is active level configurable and can accept a ready signal from an external device.
  - The serial clock is derived from the reference 13-MHz clock: The serial clock period is derived from the reference 13-MHz clock and can be configured as:

    T_{SCLK} = CK_FREQ * Csi_FRQ * T_{13M} = [2/4/7/10] * [2/4/8] * T_{13M}

    The serial clock polarity can be selected.
  - Auto-CS toggle-transmit and DMA-TX modes are supported.
- HDQ and 1-Wire master controller: The HDQ/1-Wire battery monitoring serial interface module implements the hardware protocol of the master function of the Benchmarq HDQ and the Dallas Semiconductor 1-Wire protocol. The module works off a command structure that is programmed into transmit command registers. The received data is in the receive data register. The firmware is responsible for doing the correct sequencing in the command registers. The module only implements the hardware interface layer of the protocols.
- Camera interface: Defined to support DSC function (preview + picture).

  Two ways to support camera on OMAP850:

  1) Using an external component that performs sensor acquisition plus compression (JPEG, for instance) connected to OMAP850 using a standard serial link such as I2C, μWire, USB, or SPI (all with DMA support).

  a) Using a CMOS sensor connected directly to OMAP850 using a parallel interface:

  An 8-bit CMOS sensor with scaling capability is supported on OMAP850 parallel-camera interface function.

  The following scheme is used:
  - SRGB -> RGB translation for preview: Performed by MPU
  - Image scaling: Performed by the CMOS sensor
  - Image windowing: Performed by DMA or MPU (on-the-fly during SRGB -> RGB conversion)
- Pulse-width tone modulator (PWT): This module generates a modulated frequency signal for the external buzzer. Frequency is programmable between 349 Hz and 5276 Hz with 12 half-tone frequencies per octave. The volume is also programmable.

- Pulse-width light modulator (PWL): This module allows control of the backlight of LCD and the keypad by employing a 4096-bit random sequence. This voltage-level-control technique decreases the spectral power at the modulator harmonic frequencies. The block uses a switchable 32-kHz clock that is independent of UPS.
- Serial port interface (SPI_100K): The serial port interface is a bidirectional three-line interface dedicated to the transfer of data to and from external devices offering a three-line serial interface.
- LCD controller: The LCD controller operates only in single-panel mode (dual-panel mode is not supported in this version). The panel size is programmable, and can have any width (line length) from 16 to 1024 pixels in 16-pixel increments. The number of lines is set by programming the total number of pixels in the LCD. The total video-frame size is programmable up to 1024x1024.

  The main features of the LCD controller are:

  - Encoded pixel data is stored in external memory in a frame buffer in 1-, 2-, 4-, 8-, 12- or 16-bit increments, and loaded into the LCD DMA 64-entry FIFO (16 bits per entry)
  - Programmable pixel display modes
  - Programmable display size
  - 16 grayscale levels
  - Palette allowing full logical-to-physical address mapping
  - Programmable pixel rate
  - Support for four types of displays: Passive and active color, and passive and active monochrome
  - A total of 3375 possible colors available in passive STN mode, allowing the display of any 16, 256 or 3375 colors in each frame, as well as 15 grayscale levels for monochrome screens
  - Support for any screen size up to 1024x1024 (assuming enough bandwidth is available)
  - Frame, line, and pixel clocks
  - ac-bias drive signal
  - 4-, 8-, 12-, 16- and pseudo 18 bit-per-pixel display modes
  - Patented dithering algorithm
- LCD low power controller (LLPC): The LLPC is a module between OMAP LCD controller and the external pins. It allows stopping of some signals, such as pixel clock and data lines for a period of time, thus reducing power dissipation on these lines and on the LCD panel itself.
- VLYNQ interface: This serial communications interface enables the extension of an internal CBA bus segment to one or more external physical devices. VLYNQ accomplishes this function by serializing bus transactions

in one device, transferring the serialized transaction between devices via a VLYNQ port, and deserializing the transaction in the external device.

VLYNQ ID = 0x000E for OMAP850

Main features:

- Low pin count (as few as three signals)
- No 3-state signals
  - All signals are dedicated and driven by one single device.
  - Provides significant reduction in I/O timing analysis complexity
  - Required to support high speed PHYs
- Scalable performance/support for different PHY technologies
  - 80/100 MHz and 1and 2 bits for TX and RX data
  - Linear increase in performance with any given PHY as the data port width is increased
- Simple packet based transfer protocol for memory mapped access
  - Write-request/data packet
  - Read request packet
  - Read response data packet
  - Interrupt request packet
- Symmetric operation
  - TX pins on first device connect to RX pins on second device and vice versa
  - Request packets, response packets, and flow control information are all multiplexed and sent across the same physical pins.
  - Supports both host/peripheral and peer to peer communication models
  - Able to emulate all currently used peripheral interface mechanisms
- Simple block code packet formatting (8b/10b).
- Supports In-band flow control.
  - No extra pins needed
  - Allows the receiver to momentarily throttle back the transmitter when overflow is about to occur
  - Uses special built-in code capability of block code to seamlessly interleave flow control information with user data
  - Allows system designers to balance cost of data buffering versus performance
- Supports multiple outstanding transactions
- Automatic packet formatting optimizations

- Internal loopback mode

❑ Dual-mode timer: This programmable interval 32-bit timer is required to generate a periodic interrupt, also called system clock tick, to OS. This is used to keep track of the current time and control the operation of device drivers.

The dual-mode timer main features are:

- Counter timer with compare and capture modes
- Autoreload mode
- Start-stop mode
- Programmable divider clock source
- 16-/32-bit addressing
- On-the-fly read/write registers
- Interrupts generated on overflow, compare and capture
- Interrupt enable
- Wake-up enable.
- Write posted mode
- Dedicated input trigger for capture mode and dedicated output trigger/ PWM signal
- OCP interface compatible

❑ SHA1/MD5 accelerator: The SHA1/MD5 security module provides hardware-accelerated hash functions. It can run either the SHA-1 algorithm in compliance with FIPS 180-1 standard or the MD5 message-digest algorithm developed by Rivest in 1991. Up to 2^{20}-1 bytes (1M byte) of data can be hashed in a single operation to produce a 160-bit signature in the case of SHA-1 and a 128-bit signature in the case of MD5.

❑ DES/3DES: The DES/3DES module provides hardware accelerated data encryption/decryption functions. It can run either the single DES algorithm or the triple DES algorithm in compliance with FIPS 46-3 standard. It supports ECB (electronic codebook) and CBC (cipher block chaining) modes of operation. It does not support the CFB (cipher feedback) or OFB (output feedback) modes of operation in hardware.

❑ Random number generator (RNG): The RNG module provides a true, non-deterministic noise source for the purpose of generating keys, initialization vectors (IVs) and other random number requirements. It is designed for FIPS 140-1 compliance, facilitating system certification to this security standard. It also includes built-in self-test (BIST) logic that allows testing the randomness of the module output and its compliance with FIPS 140-1 standard. An ANSI X9.17, annex C post-processor is available to meet the NIST requirements of FIPS 140-1.

1.2.2 Shared Module

In addition to the GSM and the MPU subsystems, some modules are used for GSM-S/MPU-S inter-working management:

- Traffic controller interface (TCIF): The TCIF module allows the GSM-S memory interface to access OMAP850 external memory through the MPU-S traffic controller.
- Intersystem communication register (ICR): The intersystem communication register module is a symmetrical interface between the GSM and MPU subsystems, which allows them to exchange synchronization flags. It also allows defining configuration values used by other modules (the advantage of defining them in this module is to make then accessible to both MPU and GSM subsystems).
- Configuration registers (PERSEUS25_CONF)
- TWL3016 shared port (SSPI)

1.3 Architecture

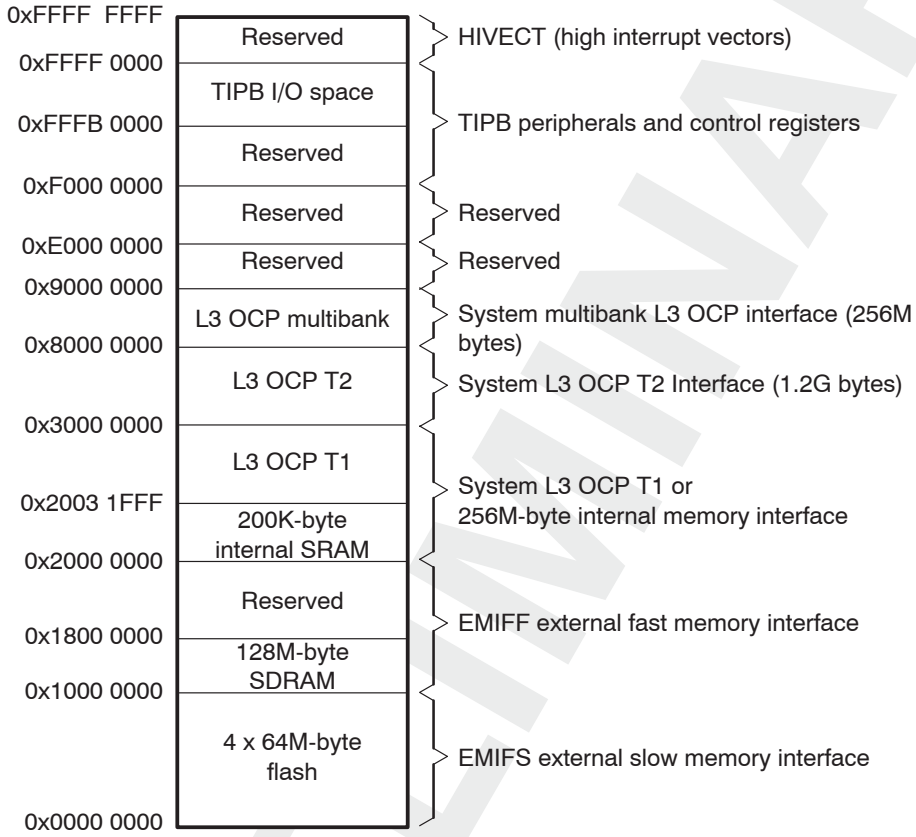
The OMAP850 device includes the MPU subsystem, the GSM subsystem, a memory and traffic controller, general-purpose peripherals, dedicated multimedia application (MMA) peripherals, and multiple interfaces. The MPU and GSM share access to the 128M bytes of fast memory space, 256M bytes of slow memory space, and 640 bytes of dual-port RAM in ICR.

1.4 Memory Maps

The MPU-S and GSM-S share the same memories thanks to a traffic controller (TC).

1.4.1 MPU-S Memory Space

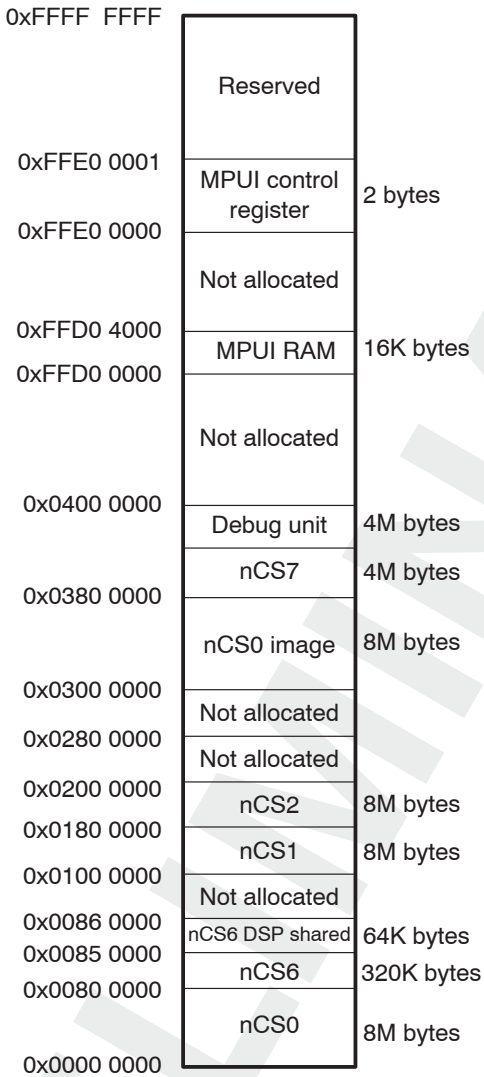
Figure 1–5. MPU-S Memory Map



1.4.2 GSM-S MPU Memory Space

The GSM-S MPU memory space is shared between the external memory interface and the TIPB bus. The Memory Interface provides six chip-select signals. All internal peripherals are mapped on GSM-S MPU memory space with a range of 32K bytes.

Figure 1−6. GSM-S Memory Map



The 8K bytes of internal RAM (0380:0000h to 0380:1000h) can overlay the first 8K-byte region 0000:0000h – 0000:1000h of the GSM-S MPU address space. In this case, the first 8K bytes of external memory are not accessible to the GSM-S MPU. This overlay is controlled by the GSM-S MPU using a register of GSM-S MPU memory interface.

1.4.3 GSM-S DSP Memory Space

The GSM-S DSP memory space (see Table 1−1) consists of the following types of memory:

- DARAM: Dual-access data RAM. It is always mapped in data space and can be overlaid in program space using the OVLY bit.
- APIRAM: Dual-access data RAM. It is always mapped in data space and can be overlaid in program space using the OVLY bit. The MPU host processor can also access this memory via the MPUI interface module.

It behaves as a communication memory between the TMS320C54x DSP and the MPU host processor.

- PROM: Program ROM, always in program space.
- DROM: Data ROM, always in data space.
- PDROM: Program or data ROM. This ROM is always mapped in program space and can also be mapped in data space by setting the DROM control bit.
- Shared PDRAM: Program/data RAM mapped on both the data space and the program space of the DSP XIO interface

The memory mapping for this S28C128 configuration is:

- 28K words of data memory (RAM-based) mapped in both data space 0 and 1.
  - 2K words of dual-access memory (DARAM)
  - 8K words of dual-access memory (API DARAM) shared between the DSP and the GSM-S MPU/DMA
  - 18K words of dual-access memory (DARAM)
- 128K words of program memory (ROM-based)
  - 100K words of program memory (PROM) mapped in program space 0
  - 20K words of data memory (DROM) mapped in data space 1
  - 8K words of mixed program/data memory (PDROM) mapped in both program space 0 and data space 1

Table 1–1 describes the GSM-S DSP memory mapping. The shaded table cells indicate memory extension on the XIO space.

Table 1–1. GSM-S DSP Memory Mapping

<table>
<tr><th></th><th>Data</th><th>Prog0</th><th>Prog1</th><th>Prog2</th><th>Prog3</th><th>Prog4</th><th>Prog5</th><th>Prog6</th></tr>
<tr><td>0000</td><td colspan="8">DARAM overlay over the program area—2K</td></tr>
<tr><td>0800
1000
1800
2000</td><td colspan="8">MPUI overlay over the program area—8K</td></tr>
<tr><td>2800
3000
3800
4000
4800
5000
5800
6000
6800</td><td colspan="8">DARAM overlay over the program area—18K</td></tr>
<tr><td>7000
7800</td><td colspan="8"></td></tr>
<tr><td>8000
8800
9000
9800
A000
A800
B000
B800
C000
C800
D000
D800
E000
E800
F000
F800</td><td rowspan="16">PDRAM
32K
Drom=0</td><td rowspan="2"></td><td rowspan="16">PROM
32K</td><td rowspan="16">PROM
32K</td><td>PROM 8K</td><td rowspan="16">PDRAM
32K</td><td rowspan="16"></td><td rowspan="16"></td></tr>
<tr><td rowspan="15">DROM
20K
Drom=1</td></tr>
<tr><td rowspan="15">PROM
28K</td><td rowspan="15"></td></tr>
<tr><td rowspan="3">PDROM
8K</td></tr>
</table>

Note: Hatched areas represent memory extension on XIO space

1.5 EDGE Feature

The GPRS-EDGE feature is added in the OMAP850.

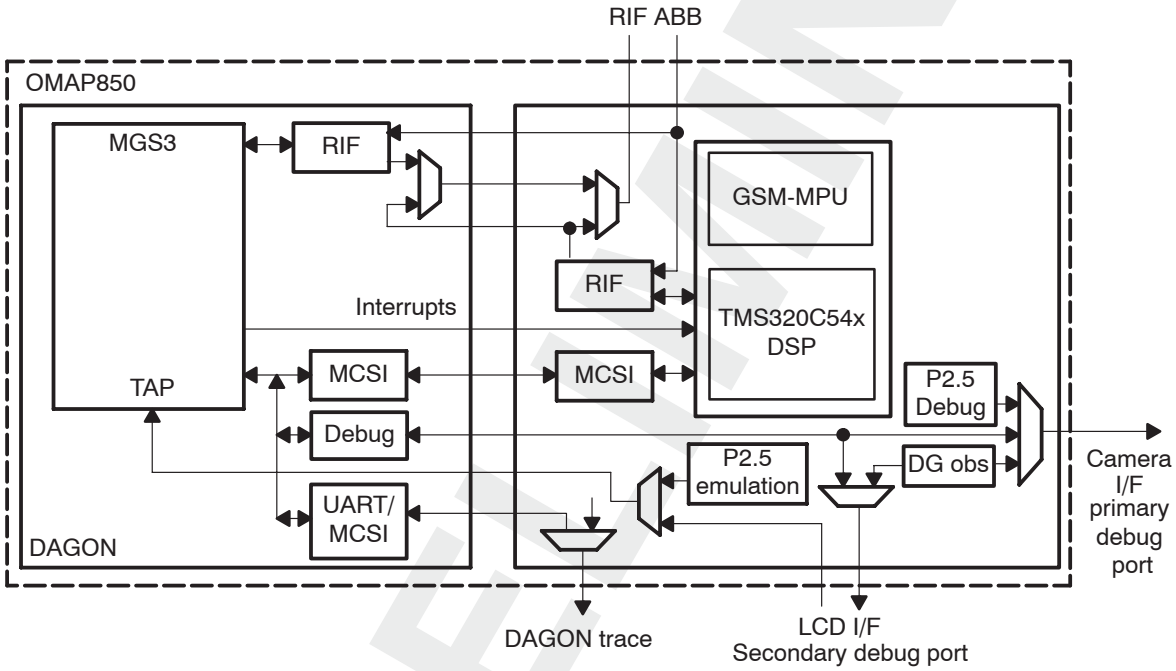
For more details on DAGON, please consult http://www.tif.ti.com/chipsetdoc/brw/browser.php?maindir=EdgeHWdoc/&directory=

1.5.1 Overview

The GPRS-EDGE feature additions are as follows (see Figure 1–7):

- ❏ Dagon mode selection
- ❏ RIF muxing
- ❏ Dagon emulation in parallel and linked to OMAP850 emulation scheme

Figure 1–7. GPRS-EDGE Feature Additions



1.5.2 New OMAP850 Configuration Mode Registers

Registers are added to control:

- ❏ Dagon mode: selection of internal GPRS-EDGE mode (INTERNAL DAGON = PERSEUS25_DAGON_MODE[0])
- ❏ Dagon emulation scheme (DG_EMU = PERSEUS25_DAGON_MODE[2:1])
- ❏ Dagon interface pull-up/down control (PERSEUS25_DAGON_IO_CONF0 and PERSEUS25_DAGON_IO_CONF1 registers).

For more detail, see Chapter 5, *Configuration Registers*.

1.5.3 DSP Memory-Mapped Registers for GPO

One new register programmable by SW is required to be mapped in DSP memory space specifically for the EDGE version.

This new peripheral is added on LEAD XIO-TIPB data space1(see Table 1–2).

Table 1–2. Data Space 1

LEAD XIO-TIPB Mapping					
Device Name		**Start Address**	**Stop Address**	**Size in Bytes**	**Data Access**
External Peripherals Mapping – I/O Space					
Strobe 3					
RIF	CS0	0000	07FF	2K	16
MCSI	CS1	0800	0FFF	2K	16
GPO	CS2	1000	17FF	2K	
Not allocated	CS3	1800	1FFF	2K	
…					

Table 1–3 describes the GPO register bits.

Table 1–3. GPO Register

Bits	Name	Function	R/W	Reset
15:3	Reserved	Reserved	R	Unknown
2	DAG_CLK13M_EN	13 MHz clock enable to DAGON 0: 13 MHz clock to DAGON is OFF 1: 13 MHz clock to DAGON is ON	R/W	1
0	RIF_MUX_CTRL	GSM/EDGE RIFs multiplexing selection 0: GSM-S RIF selected to connect to ABB 1: DAGON EDGE RIF selected to connect to ABB	R/W	1

1.5.4 DAGON Clock and Reset Scheme

Dagon reset is under control of GSM-S nreset_out (same as OMAP730, GSM-MPU memory mapped register output).

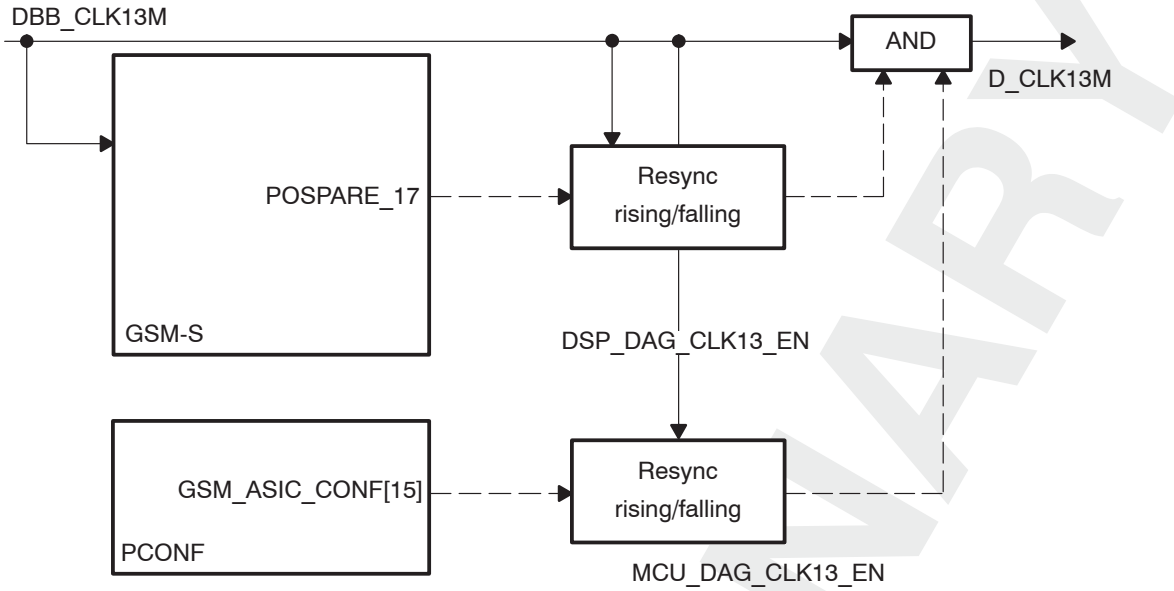
DAGON clock scheme:

13Mhz clock to Dagon can be shut-off by SW from 2 sources:

- GSM-MPU: usage of the bit DAG_CLK13M_EN = GSM_ASIC_CONF[15] in PERSEUS25_CONF module.
- DSP: usage of the bit DAG_CLK13M_EN = GPO[2].

The 13Mhz clock to Dagon enable/disable is made glitch-free (see Figure 1–8).

Figure 1–8. DAGON Trace



1.5.5 DAGON Functional Interface Handling

This includes MCSI and general interrupt lines.

- GSM MCSI
  - Outputs are always connected to DAGON ports and P2.5 pin multiplex logic
  - Inputs are connected to P2.5 pin multiplex logic when INTERNAL_DAGON = PERSEUS25_DAGON_MODE[0] = 0
  - Inputs are connected to DAGON ports when INTERNAL_DAGON = PERSEUS25_DAGON_MODE[0] = 1
  - Dagon Top pin D_MCSI_TXD => GSM_MCSI_RXD
  - Dagon Top pin D_MCSI_FSYNC (IN) => GSM_MCSI_FSYNC (IN)
  - Dagon Top pin D_MCSI_CLK (IN) => GSM_MCSI_CLK (IN)
- DAGON input interrupts (D_IN_INT[2:0])
  - Outputs of Perseus2.5 for Dagon => always connected to DAGON ports and P2.5 pin multiplex logic
  - GSM-S DSP GBO XF (LEAD XF) connected to D_IN_INT[0]
  - GSM-S TPU IT TDMA IT_FRAME connected to D_IN_INT[1]
  - GSM-S GPIO IO_GSM_9 output connected to D_IN_INT[2]
- DAGON output interrupts (D_OUT_INT[1:0])
  - Inputs of Perseus2.5 for internal Dagon => connected accorded to INTERNAL_DAGON = PERSEUS25_DAGON_MODE[0]
  - D_OUT_INT[0] connected to GSM-S External DSP IRQ EXT_DSP_NIRQ when INTERNAL_DAGON = PERSEUS25_DAGON_MODE[0] = 1 else signal from pin multiplex.
  - D_OUT_INT[1] connected to GSM-S External MCU IRQ EXT_ARM_NIRQ when INTERNAL_DAGON = PERSEUS25_DAGON_MODE[0] = 1 else signal from pin multiplex.

1.5.6 DAGON UART for Trace Handling

Dagon UART port is used only for SW trace purpose. Dagon UART is added in pin multiplex on MMC interface (see Table 1–4).

Table 1–4. DAGON UART Pin Multiplex

Top-level Pin	Mode 1 to 4	Mode 5		Mode 6
Sdmc_dat_0	Idem	DG_UART_TX	O	Idem
Sdmc_dat_1	Idem	DG_UART_RX	I	Idem
Sdmc_dat_2	Idem	DG_UART_CTS	I	Idem
Sdmc_dat_3	Idem	DG_UART_RTS	O	Idem

Chapter 2

OMAP3.2.3/OMAP3.2.4 Differences

This chapter describes the differences between OMAP3.2.3 and OMAP3.2.4.

Topic **Page**

2.1 OMAP3.2.4 Change Summary 2-2

2.2 OMAP3.2.4 Changes and Programming Model Differences 2-4

2.1 OMAP3.2.4 Change Summary

Table 2–1 summarizes differences between OMAP3.2.4 and previous revisions of OMAP3.2 core.

Table 2–1. OMAP3.2.x and OMAP3.2.4 Change Summary

Change ID	Change Name	Category	Software Compatibility	Hardware Compatibility
ECN1	32-bit frame index	New feature	Compatible with OMAP3.2 after reset	No impact
ECN2	Packet hardware synchronized transfer	New feature	Compatible with OMAP3.2 after reset	No impact
ECN3	Element counter	New feature	Compatible with OMAP3.2 after reset	No impact
ECN4	Removal of async mode logic	Feature change	Compatible after reset.	No impact
ECN5	Add window tracer lock register (WT)	New feature	Compatible with OMAP3.2 after reset	No impact
ECN6	External clock gating during asynchronous write	New feature	Compatible with OMAP3.2 after reset	No impact
ECN7	Programmable ready sampling timing	New feature	Compatible with OMAP3.2 after reset	No impact
ECN8	Synchronous BURST write in RDMODE 7	Performance enhancement	Compatible with OMAP3.2 after reset	Write protocol changed in MODE7
ECN9	Abort handling for reserved space	New feature	No impact	No impact
ECN11	Control FULL HAND shaking during writes	New feature	Compatible with OMAP3.2 after reset	No impact
ECN13	TC doubler	New feature	New register added	No impact
ECN14	ac parameter optimization	Performance enhancement	New register bit added	No impact
ECN15	Request to CAS extra cycle reduction	Performance enhancement	Compatible OMAP3.2 at reset	No impact
ECN16	Back-to-back single write access from same host	Performance enhancement	Compatible OMAP3.2 at reset	No impact
ECN17	Trp reduction from 5 to 4 cycles	Performance enhancement	Compatible OMAP3.2 at reset	No impact
ECN 18	Back-to-back single read access from same host	Performance enhancement	Compatible OMAP3.2 at reset	No impact
ECN19	TWR removal for different banks	Performance enhancement	Compatible OMAP3.2 at reset	No impact
ECN20	DMA 32-byte FIFO	Performance enhancement	Compatible OMAP3.2 at reset	No impact
ECN24	External flash clock balancing	Timing closure improvement	Compatible with OMAP3.2 after reset	No impact

Table 2–1. OMAP3.2.x and OMAP3.2.4 Change Summary (Continued)

Change ID	Change Name	Category	Software Compatibility	Hardware Compatibility
ECN27	Dynamic power saving mode	Performance (power) improvements.	Compatible with OMAP3.2 after reset	No impact
ECN28	Clock gating logic depending on master command (MCMD) for OCPI.	Performance (power) improvements.	Compatible with OMAP3.2 after reset	No impact
BTS12	LCD 1024 line interrupt	Bug fix	No impact	No impact
BTS13	OMAP3.1 data alignment issue	Bug fix	No impact	No impact

2.2 OMAP3.2.4 Register Changes and Programming Model Differences

2.2.1 ECN1: 32-Bit Frame Index (DMA)

Type of change: New feature

Interface and mode impacted: DMA rotation feature

2.2.1.1 Enhancement

OMAP3.2.4 implements a DMA 2-D transfer feature, with frame index size increasing from 16 bits to 32 bits for double-index address mode to improve DMA rotation feature.

2.2.1.2 Register List/Delta Registers

The added 32-bit register for source frame index is the DMA channel extended source frame index (DMA_CEXSFI) and contains the following two 16-bit registers:

- DMA_CEXSFI_U (DMA channel extended source frame index upper—see Table 2–3)
- DMA_CEXSFI_L (DMA channel extended source frame index lower—see Table 2–4)

The added 32-bit register for destination frame index is the DMA_CEXDFI U (DMA channel extended destination frame index) and contains the following two 16-bit registers:

- DMA_CEXDFI_U (DMA channel extended destination frame index upper—see Table 2–5)
- DMA_CEXDFI_L (DMA channel extended destination frame index lower—see Table 2–6)

These registers can be accessed with 16-bit access only.

A new bit (DMA_CCR2[7]) called FRMINDEX_32BIT has been added, which allows configuration of the frame index size (16-bit or 32-bit).

2.2.1.3 Programming Model

A new logical-channel-based programming model is introduced in the OMAP3.2.4 DMA. To enable OMAP 3.2 mode, the OMAP_31_MAPPING_DISABLE bit must be set to 1 in the DMA_GSCR register. The detailed programming model is shown in Table 2–2:

Table 2–2. DMA Channel Control Register 2 (DMA_CCR2)

Base Address = 0xFFFE:D800, Offset = 0x24 + n x 0x40				
Bit	**Name**	**Function**	**Type**	**Reset**
15:9	RESERVED	Reserved	R/W	Undefined
8	Packet_Size_CSFI_CDFI	When using packet synchronized transfer mode, it is possible to select between the DMA_CSFI and DMA_CDFI registers for choosing the number of elements in a packet transfer (or subframe). If DMA_CCR2[8] = 1, then the number of elements in the packet is defined in DMA_CSFI[15:0] If DMA_CCR2[8] = 0, then the number of elements in the packet is defined in DMA_CDFI[15:0] If Number_of_Element_in_packet = 0, then DMA hardware automatically assigns packet size to 1. The value in Packet_Size_CSFI_CDFI bit is valid only under the following conditions: - DMA_CCR.OMAP_3_1_ COMPATIBLE_DISABLE = 1 - DMA_CCR.SYNC ≠ 0000 - DMA_CCR.SYNC_PR ≠ 0 - DMA_CCR.FS = 1 - DMA_CCR2.BS = 1	R/W	Undefined
7	Frmindex_32bit	0: Frame index is 16-bit: to use the 16-bit frame indices (source and destination), ensure that the FRMINDEX_32BIT bit is cleared, and set the DMA_CSFI and DMA_CDFI registers appropriately. 1: Frame index is 32-bit: to use the 32-bit indices, set the OMAP_3_1_COMPAT-IBLE_DISABLE to 1 in the DMA_CCR register, set the FRMINDEX_32BIT bit to 1, and set the DMA_CEXSFI_U, DMA_CEX-SFI_L, DMA_CEXDFI_U, and DMA_CEXDFI_L regsiters appropriately	R/W	Undefined
6:3	Reserved	Reserved	R/W	Undefined
2	BS		R/W	Undefined
1	Transparent_Copy_Enable		R/W	Undefined
0	Constant_Fill_Enable		R/W	Undefined

Note: One bit is added in DMA channel control register 2 (DMA_CCR2) for each of the 16 channels.

Table 2–3. Extended Source Frame Index Registers (DMA_CEXSFI_U)

Base Address = 0xFFFE:D800, Offset = 0x38 + n x 0x40			
Bit	**Name**	**Type**	**Reset**
15:0	Extended source frame index (most significant word)	R/W	Undefined

Table 2–4. Extended Source Frame Index Registers (DMA_CEXSFI_L)

Base Address = 0xFFFE:D800, Offset = 0x3A + n x 0x40			
Bit	**Name**	**Type**	**Reset**
15:0	Extended source frame index (least significant word)	R/W	Undefined

IF OMAP_31_MAPPING_DISABLE = 0 (OMAP3.1 mode)

DMA_CSFI[15:0]: defines the source frame index.

ELSE OMAP_31_MAPPING_DISABLE =1 (OMAP3.2 compatible mode)

IF DMA_CCR2.Frmindex_32bit = 0 (FRAME_INDEX of DMA logical channel is 16 bits wide)

DMA_CSFI[15:0]: defines the source frame index.

ELSE (FRAME_INDEX of DMA logical channel is 32 bits wide)

DMA_CEXSFI_L[15:0]: defines the lower 16 bits of the source frame index.

DMA_CEXSFI_U[15:0]: defines the upper 16 bits of the source frame index.

Table 2–5. Extended Destination Frame Index Registers (DMA_CEXDFI_U (0xFFFE:D83C))

Base Address = 0xFFFE:D800, Offset = 0x3C + n x 0x40			
Bit	**Name**	**Type**	**Reset**
15:0	Extended destination frame index (most significant word)	R/W	Undefined

Table 2–6. Extended Destination Frame Index Registers (DMA_CEXDFI_L)

Base Address = 0xFFFE:D800, Offset = 0x3E + n x 0x40			
Bit	**Name**	**Type**	**Reset**
15:0	Extended destination frame index (least significant word)	R/W	Undefined

IF OMAP_31_MAPPING_DISABLE = 0 (OMAP3.1 mode)

DMA_CDFI[15:0]: defines the destination frame index.

ELSE OMAP_31_MAPPING_DISABLE =1 (OMAP3.2 compatible mode)

IF DMA_CCR2.FRMINDEX_32bit = 0 (Frame_Index of DMA logical channel is 16 bits wide)

DMA_CDFI[15:0]: defines the destination frame index.

ELSE (FRAME_INDEX of DMA logical channel is 32 bits wide)

DMA_CEXDFI_L[15:0]: defines the lower 16 bits of the destination frame index.

DMA_CEXDFI_U[15:0]: defines the upper 16 bits of the destination frame index.

2.2.1.4 Possible Effect on Existing OMAP3.2 Software

There is no change required for software for OMAP3.2 compatibility. Setting the FRAME_32BIT field in DMA CCR2 register can enable this feature.

2.2.2 ECN2: Packet Hardware Synchronized Transfer (DMA)

Type of change: New feature

Interface and mode impacted: DMA synchronized transfer

2.2.2.1 Enhancement

This packet synchronized transfer is useful for large transfers. If N elements must be transferred for each request, then with frame synchronized transfers one logical channel can transfer as many as N*2^16 elements. However, for packet synchronized transfers the total transfer size is equal to 2^16 * 2^16 / N.

2.2.2.2 Register List/Delta Registers

A new bit (DMA_CCR2[8]) called PACKET_SIZE_CSFI_CDFI has been added that selects either the SOURCE_FRAME_INDEX_REGISTER or DESTINATION_FRAME_INDEX_REGISTER to determine the number of elements in packet to transfer.

2.2.2.3 Programming Model

A new logical-channel-based programming model is introduced into OMAP3.2.4 DMA programming model. To enable OMAP 3.2 mode, the OMAP_3_1_COMPATIBLE_DISABLE bit must be set to 1 in the DMA_CCR register. The new added bit (DMA_CCR2[8]) called PACKET_ SIZE_CSFI_CDFI allows selection of either the SOURCE_FRAME_ INDEX_REGISTER or the DESTINATION_FRAME_INDEX_REGISTER to be used to set the number of element in packet to transfer. The detailed programming model is shown Table 2–7.

Table 2–7. DMA Channel Control Register 2 (DMA_CCR2)

Base Address = 0xFFFE:D800, Offset = 0x24 + n x 0x40				
Bit	**Name**	**Function**	**Type**	**Reset**
15:9	RESERVED	Reserved	R/W	Undefined
8	PACKET_SIZE_ CSFI_CDFI	When using packet synchronized transfer mode, it is possible to select between the DMA_CSFI and DMA_CDFI registers for choosing the number of elements in a packet transfer (or subframe). If DMA_CCR2[8] = 1, then the number of element in packet is defined in DMA_CSFI[15:0] If DMA_CCR2[8] = 0, then the number of elements in the packet is defined in DMA_CDFI[15:0] If the number of elements in the packet = 0, then DMA hardware automatically assigns packet size to 1. The value in PACKET_SIZE_CSFI_CDFI bit is valid only under the following conditions: - DMA_CCR.OMAP_3_1_ COMPATIBLE_DISABLE = 1 - DMA_CCR.SYNC ≠ 0000 - DMA_CCR.SYNC_PR ≠ 0 - DMA_CCR.FS = 1 - DMA_CCR2.BS = 1	R/W	Undefined
7	FRMINDEX_32bit	0: Frame index is 16-bit: to use the 16-bit frame indices (source and destination), ensure that the FRMINDEX_32BIT bit is cleared, and set the DMA_CSFI and DMA_CDFI registers appropriately. 1: Frame index is 32-bit: to use the 32-bit indices, set the OMAP_3_1_COMPATIBLE_DISABLE to 1 in the DMA_CCR register, set the FRMINDEX_32BIT bit to 1, and set the DMA_CEXSFI_U, DMA_CEXSFI_L, DMA_CEXDFI_U, and DMA_CEXDFI_L regsiters appropriately	R/W	Undefined
6:3	Reserved	Reserved	R/W	Undefined
2	BS		R/W	Undefined
1	TRANSPARENT_ COPY_ENABLE		R/W	Undefined
0	CONSTANT_FILL_ ENABLE		R/W	Undefined

Note: One bit is added in DMA channel control register 2 (DMA_CCR2) for each of the 16 channels.

2.2.2.4 Possible Effect on Existing OMAP3.2 Software

There is no change required for software for OMAP3.2 compatibility. Setting both BS bit of CCR2 register and FS bit of CCR register to1 enables this feature and sets the transfer size to be packet synchronized transfer.

2.2.3 ECN3: Element Counter (DMA)

Type of change: New feature

Interface and mode impacted: Status of channel transfer (DMA)

2.2.3.1 Enhancement

If the total number of elements is more than 2^16, then CASC and CDAC counters are not effective (as they are 16-bit counters). For such transfers this 32-bit element counter can be used.

2.2.3.2 Register List/Delta Registers

A 32-bit element counter called DMA_CCDEN (channel current destination element number) has been added for each logical channel and contains the following two 16-bit registers:

- DMA_CCDEN_L (DMA channel current destination element number lower—see Table 2–8)
- DMA_CCDEN_U (DMA channel current destination element number upper—see Table 2–9)

It can be accessed by 2 X 16-bit accesses only.

2.2.3.3 Programming Model

Table 2–8. DMA Channel Current Destination Element Number Lower (DMA_CCDEN_L)

Base Address = 0xFFFE:D800, Offset = 0x34 + n x 0x40			
Bit	**Name**	**Type**	**Reset**
15:0	CCDEN_L	R	Undefined

Table 2–9. DMA Channel Current Destination Element Number Upper (DMA_CCDEN_U)

Base Address = 0xFFFE:D800, Offset = 0x36 + n x 0x40			
Bit	**Name**	**Type**	**Reset**
15:0	CCDEN_U	R	Undefined

The DMA_CCDEN can be read on the fly. In case of 16-bit access, the DMA_CCDEN_L register should be read first.

2.2.3.4 Possible Effect on Existing OMAP3.2 Software

There is no change required for software for OMAP3.2 compatibility. These registers are read-only and can be read to determine the number of elements transferred.

2.2.4 ECN4: Removal of Async Mode Logic (CLKRST)

Type of change: Feature change

Interface and mode impacted: Clock modes of OMAP3.2 (by default compatible)

2.2.4.1 Enhancement

In the current implementation, the sync module transfers data from one clock domain (MPU/ DSP) to another clock domain (TC). In the current implementation, there are three types of relationship between the two clocks, either synchronous, synchronous scalable, or asynchronous. But in OMAP 3.2.4 only one DPLL is used, so the possible combination between the two clocks are only synchronous and synchronous scalable. Logic for the asynchronous mode can be removed.

2.2.4.2 Register List/Delta Registers

No change

2.2.4.3 Programming Model

No change

2.2.4.4 Possible Effect on Existing OMAP3.2 Software

By default (at reset) the behavior is compatible with OMAP3.2.

2.2.5 ECN5: Add Window Tracer Lock Register (WT)

Type of change: New feature

Interface and mode impacted: Window tracer register writes (WT)

2.2.5.1 Enhancement

The window tracer (WT) module is used to capture the memory transactions from four memory interfaces:

- EMIFS
- EMIFF
- OCP-T1
- OCP-T2

Up to two WTs can be used in each memory interface. Each WT has two 32-bit boundary physical address registers, one for the window top address and the other for the window bottom address.

2.2.5.2 Register List/Delta Registers

A window tracer lock register WT_LOCK is added (see Table 2–10).

2.2.5.3 Programming Model

Table 2–10. WT Window Tracer Lock (WT_LOCK)

Base Address = 0xFFFE:D500, Offset = 0x44				
Bit	**Name**	**Function**	**Type**	**Reset**
31:1	RESERVED	Read is undefined. Write must be zero.	R/W	Undefined
0	LOCK	Setting 1 to this bit field locks all window tracer register values. Lock bit can be reset by only a warm or cold reset.	R/W	0

Writing 1 to WT_LOCK register bit[0] locks all window tracer registers values. Any following write access to the window tracer registers does not update registers value even though the write transaction is completed successfully. The WT_LOCK bit[0] lock bit can not be reset outside a warm or cold reset.

2.2.5.4 Possible Effect on Existing OMAP3.2 Software

By default (at reset) the behavior is compatible with OMAP3.2.

2.2.6 ECN6: External Clock Gating During Asynchronous Write (EMIFS)

Type of change: New feature

Interface and mode impacted: RDMODE 4 and 5 of EMIFS

2.2.6.1 Enhancement

In the current OMAP3.2 EMIFS implementation, the external clock (flash clock) provided to the synchronous device is toggling during write access when the associated CS has been programmed to support synchronous read access (RDMODE 4 and 5) even though the write access is an asynchronous write access and that flash clock is not used by the memory device. The toggling of the flash clock device during write accesses may have or may not have impact depending on flash memory vendors and device type. Sensitive device current implementation requires cautious EMIFS CS RDMODE programming change (synchronous to asynchronous mode) and flash device configuration change before flash programming can proceed.

2.2.6.2 Register List/Delta Registers

To enhance read while write flexibility, the EMIFS flash clock device during write access in RMODE 4 and 5 is controlled. A CLKMASK bit field is added to the advanced EMIFS configuration register to control the flash clock during the write operation. If the bit field is set to 1, EMIFS keeps the flash clock at low level during the whole write access. Apart from flash clock gating, the write access behavior is unchanged from current OMAP3.2 EMIFS implementation. EMIFS flash clock behavior is not modified from the current OMAP3.2 EMIFS implementation for any RMODE value other than 4 and 5. Reset value of CLKMASK bit field is 0; this EMIFS flash clock behavior is compatible with the current OMAP3.2 EMIFS implementation.

2.2.6.3 Programming Model

At reset, CLKMASK is 0 to maintain compatability with legacy software for OMAP3.2. The detailed programming model is as follows:

CLKMASK bit field is added in every advanced CS configuration register (see Table 2−11):

EMIFS advanced CS nCS0 configuration register FLASH_ACFG_0 (0xFFFE:CC50)

EMIFS advanced CS nCS1 configuration registers FLASH_ACFG_1 (0xFFFE:CC54)

EMIFS advanced CS nCS2 configuration registers FLASH_ACFG_2 (0xFFFE:CC58)

EMIFS advanced CS nCS3 configuration registers FLASH_ACFG_3 (0xFFFE:CC5C)

Table 2−11. EMIFS Advanced Chip-Select Configuration FLASH_ACFG_n

Base Address = 0xFFFE:CC00, Offset = 0x50 + n x 4				
Bit	**Name**	**Function**	**Type**	**Reset**
31:11	RESERVED	Reserved		
10	CLKMASK	0: Flash clock is toggling during write operations for RDMODE 4 and 5. 1: Flash clock is driven low during write operations for RDMODE 4 and 5.	R/W	0
9	BTMODE		R/W	0
8	ADVHOLD		R/W	0
7:4	OEHOLD		R/W	0
3:0	OESETUP		R/W	0

2.2.6.4 Possible Effect on Existing OMAP3.2 Software

By default (at reset) the behavior is compatible with OMAP3.2. Setting CLKMASK bit field in the associated advanced CS configuration registers can enable this feature.

2.2.7 ECN7: Programmable Ready Sampling Timing (EMIFS)

Type of change: New feature

Interface and mode impacted: RDMODE 4 and 5 of EMIFS

2.2.7.1 Enhancement

The external READY pin is monitored by the EMIFS when the associated CS is programmed in full handshaking mode to enable dynamic handshaking with the external flash memory device. The READY pin is monitored synchronously to the EMIFS flash clock when the associated CS is programmed with RMODE 4 and 5. In the current OMAP3.2 implementation, the READY pin must be valid (asserted or de-asserted) one flash clock cycle a head of the data phase that is supposed to stall (not READY) or to acknowledge (READY). Some flash memory does not support this one cycle READY pipeline scheme and is limited to a scheme where READY is valid in same cycle as the data phase that

is supposed to stall (not READY) or to acknowledge (READY). To enhance the READY monitoring scheme READY can be considered valid one cycle a head, or in the same cycle as the data phase that is supposed to handshake with the external flash memory device.

Setting READY_CONFIG bit field in associated advanced CS configuration registers can enable this feature. This feature can be used with AMD muxed memories, which support RDY assertion during valid data.

2.2.7.2 Register List/Delta Registers

A READY_CONFIG bit field is added in all advanced EMIFS chip-select configuration registers to control the READY pin monitoring with respect to flash clock cycle.

The READY pin monitoring behavior is not modified for any other RDMODE configuration than RMODE 4 and 5. By default this register is set to 0 (OMAP3.2 compatibility).

2.2.7.3 Programming Model

At reset, CLKMASK is 0 to maintain compatability with legacy software for OMAP3.2. The detailed programming model is as follows:

Advanced CS Configuration Register

The READY_CONFIG bit is added in every advanced CS configuration registers (see Table 2–12):

EMIFS advanced CS nCS0 configuration register FLASH_ACFG_0 (0xFFFE:CC50)

EMIFS advanced CS nCS1 configuration registers FLASH_ACFG_1 (0xFFFE:CC54)

EMIFS advanced CS nCS2 configuration registers FLASH_ACFG_2 (0xFFFE:CC58)

EMIFS advanced CS nCS3 configuration registers FLASH_ACFG_3 (0xFFFE:CC5C)

Table 2–12. EMIFS Advanced Chip-Select Configuration FLASH_ACFG_n

Base Address = 0xFFFE:CC00, Offset = 0x50 + n x 4				
Bit	**Name**	**Function**	**Type**	**Reset**
31:12	RESERVED			
11	READ_CONFIG	0: Ready is monitored 1 clock cycle ahead of the data phase 1: Ready monitored in same cycle than the data phase	R/W	0
10	CLKMASK		R/W	0
9	BTMODE		R/W	0
8	ADVHOLD		R/W	0

Table 2–12. EMIFS Advanced Chip-Select Configuration FLASH_ACFG_n (Continued)

Base Address = 0xFFFE:CC00, Offset = 0x50 + n x 4				
Bit	**Name**	**Function**	**Type**	**Reset**
7:4	OEHOLD		R/W	0
3:0	OESETUP		R/W	0

2.2.7.4 Possible Effect on Existing OMAP3.2 Software

There is no change required for software for OMAP3.2 compatibility. Setting READY_CONFIG bit field in associated advanced CS configuration registers can enable this feature.

2.2.8 ECN8: Synchronous BURST Write in RDMODE 7 (EMIFS)

Type of change: Performance enhancement

Interface and mode impacted: RDMODE 7 of EMIFS

2.2.8.1 Enhancement

In the current EMIFS implementation, the burst writes access (4 x word32) is split into four sequential single word32 write accesses if the associated CS is programmed in RDMODE 7 (internal ASIC RAM protocol). To improve write burst performance, implement a true write burst mode protocol if the associated CS is programmed in RDMODE 7.

2.2.8.2 Register List/Delta Registers

No change

2.2.8.3 Programming Model

No change

2.2.8.4 Possible Effect on Existing OMAP3.2 Software

Write protocol in MODE7 has been changed to increase the performance; ACE RAM protocol has been followed, and ther is no effect on software.

2.2.9 ECN9: Abort Handling for Reserved Space (WT)

Type of change: New feature

Interface and mode impacted: Debugging (WT)

2.2.9.1 Enhancement

When an access to a reserved memory location occurs, the traffic controller terminates the illegal access and generates an abort interrupt.

- ❏ If the faulting access was a read access, then a 0 value is read.
- ❏ The traffic controller raises an interrupt, sets the abort flag bit in the register RES_SPS_ATYPER, and the requested address in the abort address register RES_SPC_ADDR.

- The abort ISR must read bit 0 (RESVADD_ABORT) of the RES_SPC_ATYPER register to determine the cause of the interrupt. This bit is automatically cleared by the hardware when the register is read.

This process gives more flexibility for debugging.

2.2.9.2 Register List/Delta Registers

The RES_SPC_ATYPER (0xFFFED548) and RES_SPC_ADDR (0xFFFE:D54C) registers hold the access abort information.

A RESVADD_ABORT bit field is added in RES_SPC_ATYPER register (see Table 2–13) to indicate a reserved memory address space access abort.

2.2.9.3 Programming Model

Table 2–13. WT Abort Handling For Reserved Space (RES_SPC_ATYPER)

Base Address = 0xFFFE:D500, Offset = 0x48				
Bit	**Name**	**Function**	**Type**	**Reset**
31:1	RESERVED	Reserved. To ensure software compatiblity, reserved bit must be written to 0 and read value must be considered undefined.		0x0000 0000
0	RESVADD_ABORT	0: No abort 1: Reserved address space access abort.	R/W	0

Host type is not required to reserve space. This abort can occur with MPU accesses only.

The RES_SPC_ADDR (0xFFFE:D54C) register holds the address of the transfer that has been aborted (see Table 2–13). This is valid only when the RESVADD_ABORT bit is set.

Table 2–14.WT Abort Transfer Address (RES_SPC_ADDR)

Base Address = 0xFFFE:D500, Offset = 0x4C			
Bit	**Name**	**Type**	**Reset**
31:0	Abort Address	R	0x 1000 0000

By default this feature is enabled, and this avoids system from hanging.

2.2.9.4 Possible Effect on Existing OMAP3.2 Software

This process gives more flexibility for debugging. There is no software change required.

2.2.10 ECN11: Control Full Handshaking During Writes (EMIFS)

Type of change: New feature

2.2.10.1 Enhancement

In OMAP3.2 EMIFS implementation, both read and write access the external READY pin can control timing if the associated CS is programmed in full hand-

shaking mode. To improve handshaking protocol flexibility, READY monitoring is enabled or disabled during write access independently of the READY monitoring during read access. This enables support of synchronous read while asynchronous write capability with full handshaking protocol during the synchronous read access, and no handshaking protocol during the write access.

The read/write READY pin dissymmetric usage is required for some flash device memory (Intel flashes) that asserts READY to low (not ready) during whole asynchronous access (read and write), while this assertion has no meaning of access timing control. For this implementation, the WRRDYMASK bit field is added to the CS dynamic wait control register.

2.2.10.2 Register List/Delta Registers

For this implementation, the WRRDYMASK bit field is added to the CS dynamic wait configuration register (see Table 2–15):

- When this bit is set to 1, EMIFS always ignores the READY signal during write access.
- When this bit is set to 0, EMIFS monitors the READY signal during write access according to FULLHANDSHAKE programming mode.

At reset the value of this register bit field is 0 to be compatible with OMAP3.2.

Table 2–15. WRRDYMASK Settings

FULLHANDSHAKE	WRRDYMASK	Description
0	0	Ready is considered for both write and read operations.
0	1	Ready is not considered only for the write operations.
1	X	Ready is not considered for both write and read operations.

2.2.10.3 Programming Model

At reset, the WRRDYMASK bit is 0 to maintain compatability with legacy software for OMAP3.2. The detailed programming model is shown in Table 2–16.

Table 2–16. EMIFS Dynamic Wait State Register FL_CFG_DYN_WAIT

Base Address = 0xFFFE:CC00, Offset = 0x40				
Bit	**Name**	**Function**	**Type**	**Reset**
31:12	RESERVED	Read is undefined. Write must be zero.	R/W	Undefined
11:8	WRRDYMASK for CS0-CS3	0: Considers the ready to extend the write access timing only when full handshaking mode is enabled 1: Masks the reay signal during write operation	R/W	0
7:4	HANDSHAKE_ENABLES for CS0-CS3	1: Disables the full handshaking mode to the EMIFS	R/W	0
3:0	WAIT_STATE_ENABLES for CS0-CS3	1: Enables the dynamic wait configuration mode.	R/W	0

2.2.10.4 Possible Effect on Existing OMAP3.2 Software

No change is required for software for OMAP3.2 compatibility. By default this new bit field (WRRDYMASK) is set to 0 to be compatible with OMAP3.2.

2.2.11 ECN13: TC Doubler (EMIFF)

Type of change: New feature

Interface and mode impacted: EMIFF doubler

2.2.11.1 Enhancement

This feature improves the MPU cache line fill performance.

2.2.11.2 Register List/Delta Registers

The added 32-bit register is called the doubler enable/disable register. The base address of this new register is 0xFFFECC00 and the offset is 0x60.

2.2.11.3 Programming Model

The DOUBLER_EN bit enables/disablesthe doubler (see Table 2–17). At reset the doubler is disabled (behavior same as OMAP3.2). You must explicitly write a 1 into the register bit to enable the doubler. Similarly the bit must be cleared explicitly to disable the doubler. If the doubler enable/disable bit is reset in between the doubler access, the reset is not reflected to EMIFF until the current access is completed (all eight readys are sent). So if there is a 8x32-bit access request and the doubler is disabled during this request, the doubler request is still generated and the doubler disabled only after EMIFF sends back eight readys.

The traffic controller doubler reduces the latency (8 TC cycles) between the two 4x32-bit accesses (for 8x32-bit MPU access requests) by acting as a fifth initiator to the MIFF (among other initiators such as the MPU, system DMA and OCP-I). The traffic controller doubler does not double the access time between the MPU and the EMIFF. Also, the traffic controller doubler only speeds up accesses between the MPU and EMIFF—not with the system DMA, etc.

Table 2–17. EMIFF Enable and Disable TC Doubler Feature (EMIFF_DOUBLER_EN)

Base Address = 0xFFFE:CC00, Offset = 0x60				
Bit	**Name**	**Function**	**Type**	**Reset**
31:1	RESERVED	Read is undefined. Write must be zero.	R	0x0000 0000
0	DOUBLER_EN	This flag indicates whether the doubler is enabled or disabled. 0: Disabled (value at reset) 1: Enabled	R/W	0

2.2.11.4 Possible Effect on Existing OMAP3.2 Software

You must change the software to use the new feature.

2.2.12 ECN14: ac Parameter Optimization/SDF133 Support (EMIFF)

Type of change: Performance enhancement

Interface and mode impacted: EMIFF optimization

2.2.12.1 Enhancement

Enabling this feature improves the overall throughput of the memory controller.

2.2.12.2 Register List/Delta Registers

The NEW_SYS_FREQ bit (in OMAP3.2) controls this feature (see Table 2–20).

2.2.12.3 Programming Model

The NEW_SYS_FREQ bit (in OMAP3.2) controls this feature as follows:

- Set to 1, this bit along with the SYS_FREQ field of the SDRAM CONFIGURATION register selects the optimized ac parameter table.
- Set to 0, the SDRAMFREQ = EMIFF_SDRAM_CONFIG[25:24] selects the OMAP3.2 ac parameter table.

Depending on the NEW_SYS_FREQ bit in the EMIFF_CONFIG_REG2 register (bit 3), the ac timings are different.

Table 2–18 and Table 2–19 (OMAP850 tables) replace the ac timings in the OMAP730 TRM.

Table 2–18. EMIFF OMAP3.2 AC Timings (EMIFF_CONFIG_REG2[3] = 0)

		tRC	tRAS	tRP	tRCD	tRRD	tRFC	tWR	tDPL	tDAL	tXSNR	tXSRD	tCCD	tCKED	tPED	tCDL	tMRD	tCDLR
SDR	SDF0–100	9	3	6	3	2	9	2	2	5	9	9	1	0.5	0.5	1	2	2
	SDF1–66	5	2	3	2	2	5	2	2	4	5	5	1	0.5	0.5	1	2	2
	SDF2–33	3	2	3	2	2	3	2	2	4	3	3	1	0.5	0.5	1	2	2
	SDF3–13	3	2	2	2	2	2	2	2	4	2	2	1	0.5	0.5	1	2	2
DDR	SDF0–100	9	3	6	3	2	13	3	3	6	200	200	4	0.5	0.5	2	2	2
	SDF1–66	5	2	3	3	2	5	3	3	5	200	200	4	0.5	0.5	2	2	2
	SDF2–33	3	2	3	3	2	3	3	3	5	200	200	4	0.5	0.5	2	2	2
	SDF3–13	3	2	2	3	2	2	3	3	5	200	200	4	0.5	0.5	2	2	2
MSDR	SDF0–100	12	5	8	4	3	12	2	2	7	12	12	1	0.5	0.5	1	2	2
	SDF1–66	7	4	5	2	2	7	2	2	6	7	7	1	0.5	0.5	1	2	2
	SDF2–33	3	2	3	2	2	3	2	2	4	3	3	1	0.5	0.5	1	2	2
	SDF3–13	2	2	2	2	2	2	2	2	4	2	2	1	0.5	0.5	1	2	2
MDDR	SDF0–100	12	5	8	4	3	16	3	3	8	21	19	4	0.5	0.5	2	2	2
	SDF1–66	7	4	5	3	2	7	3	3	7	14	10	4	0.5	0.5	2	2	2
	SDF2–33	3	2	3	3	2	3	3	3	5	7	3	4	0.5	0.5	2	2	2
	SDF3–13	3	2	2	3	2	2	3	3	5	3	2	4	0.5	0.5	2	2	2

Table 2–19. EMIFF Optimized AC Timings (EMIFF_CONFIG_REG2[3] = 1)

		tRC	tRAS	tRP	tRCD	tRRD	tRFC	tWR	tDPL	tDAL	tXSNR	tXSRD	tCCD	tCKED	tPED	tCDL	tMRD	tCDLR
SDR	SDF0–133	9	6	3	3	2	9	2	0	0	9	9	1	0.5	0.5	1	2	0
	SDF1–100	7	5	2	2	2	7	2	0	0	7	7	1	0.5	0.5	1	2	0
	SDF2–66	6	4	2	2	2	6	2	0	0	6	6	1	0.5	0.5	1	2	0
	SDF3–33	3	2	2	2	2	3	2	0	0	3	3	1	0.5	0.5	1	2	2
DDR	SDF0–133	9	6	3	3	2	10	2	0	0	200	200	4	0.5	0.5	0	2	2
	SDF1–100	7	5	2	3	2	8	2	0	0	200	200	4	0.5	0.5	0	2	2
	SDF2–66	6	4	2	3	2	6	2	0	0	200	200	4	0.5	0.5	0	2	2
	SDF3–33	3	2	2	3	2	3	2	0	0	200	200	4	0.5	0.5	0	2	2
MSDR	SDF0–133	11	7	4	4	2	13	2	0	0	15	15	1	0.5	0.5	1	2	0
	SDF1–100	9	7	3	3	2	10	2	0	0	12	12	1	0.5	0.5	1	2	0
	SDF2–66	7	5	2	2	2	7	2	0	0	8	8	1	0.5	0.5	1	2	0
	SDF3–33	3	2	2	2	2	4	2	0	0	4	4	1	0.5	0.5	1	2	0
MDDR	SDF0–133	11	7	4	4	2	13	2	0	0	27	27	4	0.5	0.5	0	2	2
	SDF1–100	9	7	3	3	2	11	2	0	0	27	27	4	0.5	0.5	0	2	2
	SDF2–66	7	5	2	3	2	8	2	0	0	14	14	4	0.5	0.5	0	2	2
	SDF3–33	3	2	2	3	2	4	2	0	0	7	7	4	0.5	0.5	0	2	2

Table 2–20. Second EMIFF SDRAM Configuration Register (EMIFF_CONFIG_REG2)

Base Address = 0xFFFE:CC00, Offset = 0x3C				
Bit	**Name**	**Function**	**Type**	**Reset**
31:4	RESERVED	Read is undefined. Write must be zero.	R	0x0000 0000
3	NEW_SYS_FREQ	0: SDRAMFREQ = EMIFF_SDRAM_CONFIG[25:24] selects the OMAP3.2 ac table. 1: Indicates system frequency is greater than 100 MHz. When set, this bit along with SDRAMFREQ = EMIFF_SDRAM_CONFIG[25:24] selects the optimized ac table.	R/W	0
2	SD_AUTO_CLK	Allow controller to suspend its internal clocks when idle. The clocks are automatically re-enabled when there is an auto-refresh or host request. 0: Disable (reset) 1: Enable This bit must be set in conjunction with the CLK bit in the SDRAM configuration register to turn off the clock to the external SDRAM device.	R/W	0
1	SLFR_RESET	Place the SDRAM self-refresh when in reset (active-high).	R/W	1
0	SLFR_STBY	Place the SDRAM into self-refresh when in standby mode (active-high).	R/W	1

2.2.12.4 Programming Model for OMAP3.2 Compatibility

By default (at reset) the behavior is compatible with OMAP32. Setting NEW_SYS_FREQ bit field of SDRAM_CONFIG2 register enables this feature.

2.2.13 ECN15: Req CAS Extra Cycle Removal

Type of change: Performance improvement

Interface and mode impacted: EMIFF optimization

2.2.13.1 Enhancement

This enhancement removes an extra cycle between the host request and the command issued to the memory when autoclock gating is off.

2.2.13.2 Register List/Delta Registers

No change

2.2.13.3 Programming Model

No change

2.2.13.4 Possible Effect on Existing OMAP3.2 Software

No change

2.2.14 ECN16: Back to Back Single \Writes From Same Host (EMIFF)

Type of change: Performance improvement

Interface and mode impacted: EMIFF optimization

2.2.14.1 Enhancement

This enhancement removes four extra cycles in case of back to back single writes from same host to same bank same row or different banks open row. This enhancement is achieved by modifying the existing controller FSM.

2.2.14.2 Register List/Delta Registers

No change

2.2.14.3 Programming Model

No change

2.2.14.4 Possible Effect on Existing OMAP3.2 Software

No change

2.2.15 ECN17: tRP Reduction From 5 to 4 Cycles (EMIFF)

Type of change : Performance improvement

Interface and mode impacted: EMIFF optimization

2.2.15.1 Enhancement

See Section 2.2.12 (ECN#14).

2.2.15.2 Register List/Delta Registers

No change

2.2.15.3 Programming Model

No change

2.2.15.4 Possible effect on Existing OMAP3.2 Software

No change

2.2.16 ECN18: Back to Back Single Reads From Same Host (EMIFF)

Type of change: Performance improvement

Interface and mode impacted: EMIFF optimization

2.2.16.1 Enhancement

This enhancement removes three extra cycles in case of back to back single reads from same host to same bank same row or different banks open row. This enhancement is achieved by modifying the existing controller FSM.

2.2.16.2 Register List/Delta Registers

No change

2.2.16.3 Programming Model

No change

2.2.16.4 Possible Effect on Existing OMAP3.2 Software

No change

2.2.17 ECN19: TWR Removal for Accesses to Different Banks (EMIFF)

Type of change: Performance improvement

Interface and mode impacted: EMIFF optimization

2.2.17.1 Enhancement

This enhancement removes the dependency of the tWR parameter in the current FSM when the precharge is to a different bank from the previous write. It results in the precharge being issued early. This enhancement is achieved by modifying the controller FSM.

2.2.17.2 Register List/Delta Registers

No change

2.2.17.3 Programming Model

No change

2.2.17.4 Possible Effect on Existing OMAP3.2 Software

No change

2.2.18 ECN20: 32-bit FIFO for Each Physical Channel (EMIFF)

Type of change: New feature

Interface and mode impacted: EMIFF optimization

2.2.18.1 Enhancement

FIFO size has been increased from 31 bytes to 32 bytes. It can now accommodate two burst reads. Earlier, after one burst read and write, there were four

dead cycles. With FIFO size 32 bits, these foiur4 dead cycles have been removed. Two burst reads are followed by two burst writes without any dead cycles.

2.2.18.2 Register List/Delta Registers

No change

2.2.18.3 Programming Model

No change

2.2.18.4 Possible Effect on Existing OMAP3.2 Software

There is no change required for software for OMAP32 compatibility. These registers are read-only and indicate the number of elements transferred.

2.2.19 ECN24: External Flash Clock Balancing (EMIFS)

Type of change: Performance enhancement

2.2.19.1 Enhancement

EMIFS internally generates divided clocks supplied to the connected synchronous memories. The same clock division is used to derive an internal reference clock used by the EMIFS module. Under certain conditions on OMAP730, the internal reference clock (for EMIFS registers) and the external flash clock (for flash memory registers) can be unbalanced resulting in difficult timing closures. On OMAP850, internal logic is added to ensure these clocks are balanced (no timing issues).

2.2.19.2 Register List/Delta Registers

No change

2.2.19.3 Programming Model

No change

2.2.19.4 Possible Effect on Existing OMAP3.2 Software

There is no change required for software for OMAP32 compatibility

2.2.20 ECN27: Dynamic Power saving mode (LCD controller).

Type of change: Performance improvement

2.2.20.1 Enhancement

This feature can be used to improve the power consumption.

2.2.20.2 Register List/Delta Registers

Added the DPS_EN bit to enable/disable clock gating in LCD (see Table 2–21).

New bit has been to be added to provide flexibility to enable and disable autoclock gating and keeping backward compatibility. By default the reset value is compatible with OMAP3.2 (autoclock gating is disabled).

2.2.20.3 Programming Model

The LCD control register (LCD_CTRL_REG) contains seven bit-fields that control various functions within the LCD controller (see Table 2–21). Reserved bits return 1 when read. The DPS_EN bit is added to control the dynamic power saving mode enabled or disabled.

Table 2–21. LCD Control Register (LCD_CTRL_REG)

Base Address = 0xFFFE:C000, Offset = 0x00				
Bit	**Name**	**Function**	**Type**	**Reset**
31:27	RESERVED	Read is undefined. Write must be zero.	R	Un defined
26	ALIGNMENT_BIT	0: Misalignment between data and pixel clock 1: Alignment between data and pixel clock	R/W	0
25	DPS_EN	Dynamic power saving mode. Enables autoclock gating if enabled. 0: Autoclock gating OFF 1: Autoclock gating ON	R/W	0
24	STN_565	12 BPP (5-6-5) mode 0: Off 1: ON 16-bit data in frame buffer, but only 12 bits are dithered and sent out.	R/W	0
23	TFT_MAP	TFT alternate signal mapping 0: Output pixel data for 1, 2, 4, and 8 BPP modes must be right-aligned on lcd_pins [11:0] 1: Output pixel data for 1, 2, 4, and 8 BPP must be converted to 5, 6, 5 format and use pins [15:0] R3 R2 R1 R0 R3 G3 G2 G1 G0 G3 G2 B3 B2 B1 B0 B3	R/W	0
22	NM	Nibble mode 0: NIbble mode is disabled 1: Nibble mode is enabled	R/W	0
21:20	PLM	Palette loading mode 00: Palette and data loading, reset value 01: Palette loading 10: Data loading	R/W	0
19:12	FDD	FIFO DMA request delay Encoded value (0-255) used to specify the number of LCD controller clocks. The input FIFO DMA request from LCD controller must be disabled. The clock count starts after 16 words read in the input FIFO. Programming FDD = 00h disables this function.	R/W	0

Table 2–21. LCD Control Register (LCD_CTRL_REG) (Continued)

Base Address = 0xFFFE:C000, Offset = 0x00				
Bit	**Name**	**Function**	**Type**	**Reset**
11	PXL_GATED	Pixel gated (for TFT mode only) 0: Pixel clock toggles always 1: Pixel clock only toggles when there is valid data to display	R/W	0
10	LINE_INT_CLR_SET	Line interrupt clear select bit 0: TIPB write 0 to clear line_int bit in LCD status register (default) 1: Line_int bit in status register is cleared at the end of the line	R/W	0
9	M8B	Mono 8-bit mode 0: LCD_PIXEL_O[3:0] is used to output four pixel values to the panel each pixel clock transition. 1: LCD_PIXEL_O[7:0] is used to output eight pixel values to the panel each pixel clock transition. This bit is ignored in all other modes.	R/W	0
8	LCDBE	LCD big endian 0: Little endian operation is selected, frame/pin buffer data is arranged into individual words of memory starting with the least significant nibble, byte or half-word. 1: Big endian operation is selected, frame/pin buffer data is arranged into individual words of memory starting with the most significant nibble, byte or half-word.	R/W	0
7	LCD_TFT	LCD TFT 0: Passive or STNdisplay operation enabled, dither logic is enabled 1: Active or TFT display operation enabled, external palette and DAC required, dither logic bypassed, pin timing changes to support continuous pixel clock, output enable, VSYNC, and HSYNC	R/W	0
6	LINE_INT_MASK	Line interrupt mask (dedicated line) 0: Masks the line interrupt path to dedicated line 1: Mask not active	R/W	0
5	LINE_INT_NIRQ_MASK	Line interrupt mask (shared line) 0: Interrupt to the LCD_NIRQ is masked 1: Interupt to the LCD_NIRQ is unmasked	R/W	0
4	LOADMASK	Load mask 0: Mask out the loaded palette interrupt path to LCD_NIRQ 1: Mask not active	R/W	0
3	LCDDONEMASK	Done mask 0: Mask out the frame done (done) interrupt path to LCD_NIRQ 1: Mask not active	R/W	0

Table 2–21. LCD Control Register (LCD_CTRL_REG) (Continued)

Base Address = 0xFFFE:C000, Offset = 0x00				
Bit	**Name**	**Function**	**Type**	**Reset**
2	VSYNC_MASK	LCD VSYNC interrupt mask 0: Interrupt to the LCD_NIRQ is masked 1: Interrupt to the LCD_NIRQ is unmasked	R/W	0
1	LCDBW	LCD monochrome 0: Color operation enabled 1: Monochrome operation enabled	R/W	0
0	LCD_ENABLE	LCD controller enable 0: LCD controller disabled 1: LCD controller enabled	R/W	0

2.2.20.4 Possible Effect on Existing OMAP3.2 Software

There is no change required for software for OMAP32 compatibility. This feature is enabled/disabled by setting DPS_EN bit field in LCD control register.

2.2.21 ECN28: Adding of Clock Gating Logic Depending on MCMD (OCPI)

Type of change: Performance improvement

2.2.21.1 Enhancement

Some of the OCPI logic is not functional when MCMD is IDLE (such as secondary command (SCMD), counter, burst counter, FSM logic). So the clock to this logic can be cut off when MCMD is IDLE.

The logic that registers the input signals and the logic that deals with the MPU signal are driven by non-gated clock and rest of the logic is driven by gated clock. Even if the input delay of the MCMD were quite low, then the gated clock can drive the registers, which are used for the purpose of registering the input signals.

All the flops of OCPI are driven by l3_ocpi_ck_gated (see Figure 2–1) or l3_ocpi_ck_not_gated (see Figure 2–2). Driving flops by l3_ocpi_ck_not_gated instead of l3_ocpi_ck provides a cts_buffer in the design so the balancing is easy.

Figure 2–1. Generation of Gated Clock

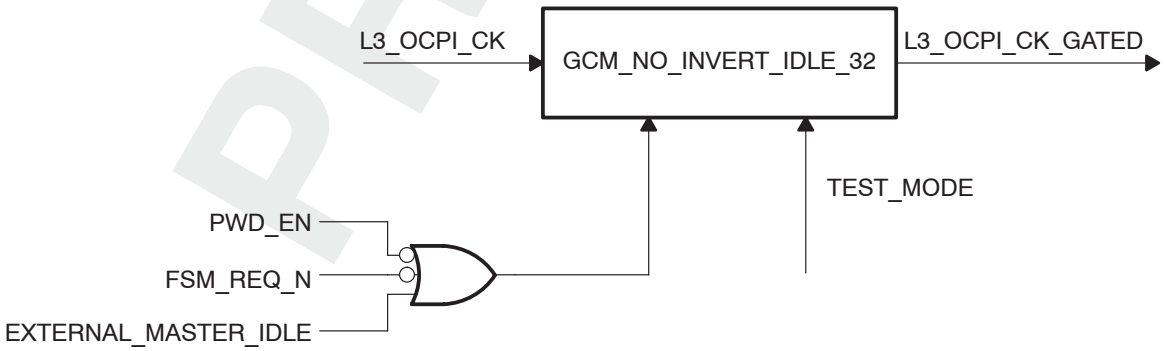
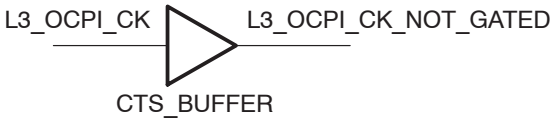


Figure 2−2. Generation of Non-Gated Clock



This feature improves power consumption.

2.2.21.2 Register List/Delta Registers

The added 32-bit register is the dynamic power-down register (see Table 2−22). The base address of this new register is 0xFFFEC320 and the offset is 0x1C.

2.2.21.3 Programming Model

The dynamic clock gating logic is software controllable through the AUTO_GATED_CLK bit:

- Setting the bit to 1 enables the autoclock gating which saves power.
- Setting the bit to 0 disables the autoclock gating.

By default the autoclock gating is disabled (bit set to 0).

Table 2–22. OCPI Dynamic Power Down (DYNAMIC_POWER_DOWN (0xFFFE:C33C))

Base Address = 0xFFFE:C320, Offset = 0x1C				
Bit	**Name**	**Function**	**Type**	**Reset**
31:1	RESERVED	Read is undefined. Write must be zero.	R	0x0000 0000
0	AUTO_GATED_CLK	0: Autoclock gating is disabled 1: Autoclock gating is enabled to save power	R/W	0

2.2.21.4 Possible Effect on Existing OMAP3.2 Software

There is no change required for software for OMAP32 compatibility. To be compatible with OMAP3.2, the reset value is 0, which disables clock gating. You can enable/disable this feature through the AOTO_GATED_CLK bit (see Table 2–22).

2.2.22 BTS12: 1024 Line Interrupt

Type of change: Bug fix

2.2.22.1 Register List/Delta Registers

The width of the LCD display status register is changed from 10 to 11 (see Table 2–23). The reset value of the register is changed from 3ff to 7ff.

2.2.22.2 Programming Model

Table 2–23. LCD Display Status Register (DISPLAY_STATUS_REGISTER (0xFFFE:C01C))

Base Address = 0xFFFE:C000, Offset = 0x1C				
Bit	**Name**	**Function**	**Type**	**Reset**
31:11	RESERVED	Read is undefined. Write must be one.	R	0x1F FFFF
10:0	CURRENT_LINE_ NUMBER	Line number being displayed. As the number of line can be programmed from 1 to 1024, the current line number varies between 0 and 1023.	R	0x7FF

2.2.22.3 Possible Effect on Existing OMAP3.2 Software

There is no change required for software for OMAP3.2 compatibility.

2.2.23 BTS13: OMAP31 Silicon Failure

Type of change: Bug fix

2.2.23.1 Register List/Delta Registers

A new ALIGNMENT_BIT bit is added in the LCD control register (see Table 2–24) to enable/disable data alignment.

2.2.23.2 Programming Model

Table 2–24. LCD Control Register (LCD_CTRL_REG)

Base Address = 0xFFFE:C000, Offset = 0x00				
Bit	**Name**	**Function**	**Type**	**Reset**
31:27	RESERVED	Read is undefined. Write must be zero.	R	Un defined
26	ALIGNMENT_BIT	0: Misalignment between data and pixel clock (compatible with OMAP3.1) 1: Alignment between data and pixel clock	R/W	0
25	DPS_EN	Dynamic power saving mode. Enables autoclock gating if enabled. 0: Autoclock gating OFF 1: Autoclock gating ON	R/W	0
24	STN_565	12 BPP (5-6-5) mode 0: Off 1: ON 16-bit data in frame buffer, but only 12 bits are dithered and sent out.	R/W	0
23	TFT_MAP	TFT Alternate Signal Mapping 0: Output pixel data for 1, 2, 4, and 8 BPP modes is right-aligned on lcd_pins [11:0] 1: Output pixel data for 1, 2, 4, and 8 BPP is converted to 5, 6, 5 format and use pins [15:0] R3 R2 R1 R0 R3 G3 G2 G1 G0 G3 G2 B3 B2 B1 B0 B3	R/W	0
22	NM	Nibble Mode 0: NIbble mode is disabled 1: Nibble mode is enabled	R/W	0
21:20	PLM	Palette Loading Mode 00: Palette and data loading, reset value 01: Palette loading 10: Data loading	R/W	0
19:12	FDD	FIFO DMA Request Delay Encoded value (0-255) used to specify the number of LCD controller clocks. The input FIFO DMA request from LCD controller should be disabled. The clock count starts after 16 words read in the input FIFO. Programming FDD = 00h disables this function.	R/W	0

Table 2–24. LCD Control Register (LCD_CTRL_REG) (Continued)

Base Address = 0xFFFE:C000, Offset = 0x00				
Bit	**Name**	**Function**	**Type**	**Reset**
11	PXL_GATED	Pixel gated (for TFT mode only) 0: Pixel clock toggles always 1: Pixel clock only toggles when there is valid data to display	R/W	0
10	LINE_INT_CLR_SET	Line interrupt clear select bit 0: TIPB write 0 to clear line_int bit in LCD status register (default) 1: Line_int bit in status register is cleared at the end of the line	R/W	0
9	M8B	Mono 8-bit mode 0: LCD_PIXEL_O[3:0] is used to output four pixel values to the panel each pixel clock transition. 1: LCD_PIXEL_O[7:0] is used to output eight pixel values to the panel each pixel clock transition. This bit is ignored in all other modes.	R/W	0
8	LCDBE	LCD big endian 0: Little endian operation is selected, frame/pin buffer data is arranged into individual words of memory starting with the least significant nibble, byte or half-word. 1: Big endian operation is selected, frame/pin buffer data is arranged into individual words of memory starting with the most significant nibble, byte or half-word.	R/W	0
7	LCD_TFT	LCD TFT 0: Passive or STN display operation enabled, dither logic is enabled 1: Active or TFT display operation enabled, external palette and DAC required, dither logic bypassed, pin timing changes to support continuous pixel clock, output enable, vsync, and hsync	R/W	0
6	LINE_INT_MASK	Line interrupt mask (dedicated line) 0: Masks the line interrupt path to dedicated line 1: Mask not active	R/W	0
5	LINE_INT_NIRQ_MASK	Line interrupt mask (shared line) 0: Interrupt to the LCD_NIRQ is masked. 1: Interupt to the LCD_NIRQ is unmasked.	R/W	0
4	LOADMASK	Load mask 0: Mask out the loaded palette interrupt path to LCD_NIRQ 1: Mask not active	R/W	0
3	LCDDONEMASK	Done mask 0: Mask out the frame done (done) interrupt path to LCD_NIRQ 1: Mask not active	R/W	0

Table 2–24. LCD Control Register (LCD_CTRL_REG) (Continued)

Base Address = 0xFFFE:C000, Offset = 0x00				
Bit	**Name**	**Function**	**Type**	**Reset**
2	VSYNC_MASK	LCD VSYNC interrupt mask 0: Interrupt to the LCD_NIRQ is masked 1: Interrupt to the LCD_NIRQ is unmasked	R/W	0
1	LCDBW	LCD monochrome 0: Color operation enabled 1: Monochrome operation enabled	R/W	0
0	LCD_ENABLE	LCD controller enable 0: LCD controller disabled 1: LCD controller enabled	R/W	0

2.2.23.3 Possible Effect on Existing OMAP3.2 Software

Existing software is not impacted by this feature. There is no change required for software for OMAP3.2 compatibility.

2.2.24 OMAP3.2.4 Registers

Changed registers or bits are in green shading.

2.2.24.1 DMA Controller Registers

See ECN1 (Section 2.2.1), ECN2 (Section 2.2.2), and ECN3 (Section 2.2.3) for changes.

Table 2–25 lists the DMA controller registers. Table 2–26 through Table 2–80 describe the regsiter bits.

Table 2–25. DMA Controller Registers

Base Address = 0xFFFE:D800				
Bits	**Register**	**Description**	**R/W**	**Offset**
16	DMA_CSDP_CHx	Channel source destination parameters	R/W	0x000 + n x 0x40
16	DMA_CCR_CHx	Channel control	R/W	0x002 + n x 0x40
16	DMA_CICR_CHx	Channel interrupt control	R/W	0x004 + n x 0x40
16	DMA_CSR_CHx	Channel status	R/W	0x006 + n x 0x40
16	DMA_CSSA_L_CHx	Channel source start address, lower bits	R/W	0x008 + n x 0x40
16	DMA_CSSA_U_CHx	Channel source start address, upper bits	R/W	0x00A + n x 0x40
16	DMA_CDSA_L_CHx	Channel destination start address, lower bits	R/W	0x00C + n x 0x40
16	DMA_CDSA_U_CHx	Channel destination start address, upper bits	R/W	0x00E + n x 0x40
16	DMA_CEN_CHx	Channel element number	R/W	0x010 + n x 0x40
16	DMA_CFN_CHx	Channel frame number	R/W	0x012 + n x 0x40
16	DMA_CFI_CHx	Channel frame index	R/W	0x014 + n x 0x40
16	DMA_CEI_CHx	Channel element index	R/W	0x016 + n x 0x40
16	DMA_CPC_CHx	Channel progress counter	R/W	0x018 + n x 0x40
16	DMA_CDAC_CHx	Channel destination address counter	R/W	0x01A + n x 0x40
16	DMA_CDEI_CHx	Channel destination element index	R/W	0x01C + n x 0x40
16	DMA_CDFI_CHx	Channel destination frame index	R/W	0x01E + n x 0x40
16	DMA_COLOR_L_CHx	Color parameter, lower bits	R/W	0x020 + n x 0x40
16	DMA_COLOR_U_CHx	Color parameter, uppar bits	R/W	0x022 + n x 0x40

Table 2–25. DMA Controller Registers (Continued)

Base Address = 0xFFFE:D800				
Bits	**Register**	**Description**	**R/W**	**Offset**
16	DMA_CCR2_CHx	Channel control 2	R/W	0x024 + n x 0x40
16	DMA_CLNK_CTRL_CHx	Channel link control	R/W	0x028 + n x 0x40
16	DMA_LCH_CTRL_CHx	Logical channel control	R/W	0x02A + n x 0x40
16	DMA_CDDEN_L	Channel current destination element number, lower bits	R/W	0x034 + n x 0x40
16	DMA_CDDEN_U	Channel current destination element number, upper bits	R/W	0x036 + n x 0x40
16	DMA_CEXSFI_U	Extended source frame index, upper bits	R/W	0x038 + n x 0x40
16	DMA_CEXSFI_L	Extended source frame index, lower bits	R/W	0x03A + n x 0x40
16	DMA_CEXDFI_U	Extended destination frame index, upper bits	R/W	0x03C + n x 0x40
16	DMA_CEXDFI_L	Extended destination frame index, lower bits	R/W	0x03E + n x 0x40
16	DMA_LCD_CTRL	LCD control	R/W	0x300
16	DMA_LCD_TOP_F1_L	LCD top address for frame buffer 1, lower bits	R/W	0x302
16	DMA_LCD_TOP_F1_U	LCD top address for frame buffer 1, upper bits	R/W	0x304
16	DMA_LCD_BOT_F1_L	LCD top address for frame buffer 1, lower bits	R/W	0x306
16	DMA_LCD_BOT_F1_U	LCD top address for frame buffer 1, upper bits	R/W	0x308
16	DMA_LCD_TOP_F2_L	LCD top address for frame buffer 2, lower bits	R/W	0x30A
16	DMA_LCD_TOP_F2_U	LCD top address for frame buffer 2, upper bits	R/W	0x30C
16	DMA_LCD_BOT_F2_L	LCD top address for frame buffer 2, lower bits	R/W	0x30E
16	DMA_LCD_BOT_F2_U	LCD top address for frame buffer 2, upper bits	R/W	0x310
16	DMA_GCR	Global control	R/W	0x400
16	DMA_GSCR	Software compatible	R/W	0x404
16	DMA_GRST	Software reset control	R/W	0x408
16	DMA_HW_ID	Hardware version IC number	R	0x442
16	DMA_PCHP_ID	Pch-P version ID	R	0x444
16	DMA_PCHM0_ID	Pch-M0 version ID	R	0x446
16	DMA_PCHM1_ID	Pch-M1 version ID	R	0x448
16	DMA_PCHG_ID	Pch-G version ID number	R	0x44A
16	DMA_PCHID_ID	Pch-D version ID number	R	0x44C
16	DMA_CAPS_0_U	Global DMA capability 0	R	0x44E
16	DMA_CAPS_0_L	Global DMA capability 0	R	0x450
16	DMA_CAPS_1	Global DMA capability 1	R	0x454

Table 2–25. DMA Controller Registers (Continued)

Base Address = 0xFFFE:D800				
Bits	**Register**	**Description**	**R/W**	**Offset**
16	DMA_CAPS_2	Global DMA capability 2	R	0x456
16	DMA_CAPS_3	Global DMA capability 3	R	0x458
16	DMA_CAPS_4	Global DMA capability 4	R	0x45A
16	DMA_PCHSR_P_0	Peripheral physical channel 0 status	R	0x460
16	DMA_PCHSR_M_0	Memory physical channel 0 status	R	0x480
16	DMA_PCHSR_M_1	Memory physical channel 1 status	R	0x482
16	DMA_PCHSR_D_0	Physical display channel 0 status	R	0x4C0

Table 2–26. Channel Source Destination Parameters Register (DMA_CSDP_CHX)

Base Address = 0xFFFE:D800, Offset = 0x000				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:14	DST_BURST_EN	Destination burst enable	R/W	0x00
13	DST_PACK	Destination packing	R/W	0x0
12:9	DST	Destination port	R/W	0x0000
8:7	SRC_BURST_EN	Source burst enable 00: Single access (no burst) 01: Single access (no burst) 10: Burst 4 11: Burst 8	R/W	0x00
6	SRC_PACK	Source packing 0: Source port never makes packed accesses 1: Source port makes packed accesses	R/W	0x0
5:2	SRC	Source port 0000: EMIFF 0001: EMIFS 0010: OCP_T1 0011: RHEA 0100: OCP_T2 0101: API Others: Illegal	R/W	0x0000
1:0	DATA_TYPE	Type of data moved in to channel 00: s8, 8 bits scalar 01: s16, 16 bits scalar 10: s32, 32 bits scalar 11: Illegal value	R/W	0x00

Table 2–27. Channel Control Register (DMA_CCR_CHX)

Base Address = 0xFFFE:D800, Offset = 0x002				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:14	DST_AMODE	Destination address mode 00: Constant address 01: Post incremented address 01: Single index (element index) 11: Double index (element index and frame index0)	R/W	0x00
13:12	SRC_AMODE	Source address mode 00: Constant address 01: Post incremented address 10: Single index (element index) 11: Double index (element index and frame index0)	R/W	0x00
11	END_PROG	End of programming. Allows the channel to reinitialize itself if AUTO_INIT is enabled.	R/W	0x0
10	OMAP_3_1_ COMPATIBLE_ DISABLE	OMAP3.0/3.1 channel compatibility control sets the DMA logical channel programming model to be used. By default, all DMA logical channels are in OMAP3.1 compatible mode: 0: Logical channel is in OMAP3.0/3.1 compatible mode 1: Logical channel is in OMAP3.2 compatible mode	R/W	0x0
9	REPEAT	Repetitive operation 0: Once current transfer is complete channel automatically reinitializes itself and starts a new transfer disregarding END_PROG 1: Once current transfer is complete channel automatically reinitializes itself and starts a new transfer only if END_PROG =1	R/W	0x0
8	AUTO_INIT	Auto_initialization at the end of transfer 0: Channel stops at the end of current transfer 1: Once current transfer is complete, the channel automatically reinitializes itself and starts a new transfer.	R/W	0x0
7	EN	Enable/Disable transfer in DMA Channel 0: Transfer stops and is reset 1: Transfer starts	R/W	0x0
6	PRIO	Channel priority 0: Channel has the low priority level 1: Channel has the high priority level	R/W	0x0
5	FS	Frame synchronization 0: An element is transferred each time a DMA request is made 1: An entire frame is transferred each time a DMA request is made	R/W	0x0

Table 2–27. Channel Control Register (DMA_CCR_CHX) (Continued)

Base Address = 0xFFFE:D800, Offset = 0x002				
Bit	**Name**	**Function**	**R/W**	**Reset**
4	SYNC_PR	0: Synchronization is made regarding to TIPB port 1: Synchronization is made regarding to API_RHEA port	R/W	0x0
3:0	SYNC	Synchronization control. Transfer synchronized on DMA_REQUEST{sync}, sync != 0.	R/W	0x0000

Table 2–28. Channel Interrupt CTRL Register (DMA_CICR_CHX)

Base Address = 0xFFFE:D800, Offset = 0x004				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:6	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
5	BLOCK_IE	End block interrupt enable 0: Channel does not interrupt the processor when the transfer of the block completes 1: Channel sends an interrupt to the processor when the transfer of block completes	R/W	0x0
4	LAST_IE	Last frame interrupt enable 0: Channel does not interrupt the processor when the transfer of last frame starts 1: Channel sends an interrupt to the processor when the when the transfer of last frame starts	R/W	0x0
3	FRAME_IE	Frame interrupt enable 0: Channel does not interrupt the processor when the transfer of current frame completes 1: Channel sends an interrupt to the processor when the transfer of current frame completes	R/W	0x0
2	HALF_IE	Half frame interrupt enable 0: Channel does not interrupt the processor when the transfer of first half of the current frame completes 1: Channel sends an interrupt to the processor when the synchronization event drop occurs	R/W	0x0
1	DROP_IE	Synchronization event drop interrupt enable 0: Channel does not interrupt the processor when the synchronization event drop occurs 1: Channel sends an interrupt to the processor if the channel transfer is synchronized on DMA requests and two successive DMA requests drop.	R/W	0x1
0	TOUT_IE	time-out interrupt enable 0: Channel does not interrupt the processor if a time-out error occurs 1: Channel sends an interrupt to the processor if a time-out error occurs	R/W	0x1

Table 2–29. Channel Status Register (DMA_CSR_CHX)

Base Address = 0xFFFE:D800, Offset = 0x006				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:7	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
6	SYNC	Synchronization status 0: No DMA request is in service 1: DMA request is made for this channel when it was in service	R/W	0x0
5	BLOCK	End block 0: Current transfer is not yet finished 1: Current transfer is finished	R/W	0x0
4	LAST	Last frame 0: Last frame did not start yet 1: Transfer of last frame has started	R/W	0x0
3	FRAME	Frame 0: Transfer of the current frame still in progress 1: A complete frame was transferred	R/W	0x0
2	HALF	Half 0: First half of the current frame not transferred yet 1: First half of the current frame was transferred	R/W	0x0
1	DROP	Drop 0: No event drop occured during transfer 1: Event drop occured during transfer	R/W	0x0
0	TOUT	Time-out 0: No time-out error occured in the channel 1: Time-out error occured in the channel	R/W	0x0

Table 2–30. Channel Source Start Address, Lower Bits Register (DMA_CSSA_L_CHX)

Base Address = 0xFFFE:D800, Offset = 0x008				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:0	CSSA_L	Channel source start address, lower bits. The source start address output by DMA is up to 32-bit byte address.	R/W	0x0

Table 2–31. Channel Source Start Address, Upper Bits Registers (DMA_CSSA_U_CHX)

Base Address = 0xFFFE:D800, Offset = 0x00A				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:0	CSSA_U	Channel source start address, upper bits. The sourcestart address is made of DMA_CSSA_U and DMA_CSSA_L	R/W	0x0

Table 2–32. Channel Destination Start Address, Lower Bits Register (DMA_CDSA_L_CHX)

Base Address = 0xFFFE:D800, Offset = 0x00C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:0	CDSA_L	Lower bits for the destination start addres, expressed in bytes. The destination start addresss is up to a 32-bit byte address made of the concatenation of DMA_CDSA_U and DMA_CDSA_L.	R/W	0x0

Table 2–33. Channel Destination Start Address, Upper Bits Register (DMA_CDSA_U_CHX)

Base Address = 0xFFFE:D800, Offset = 0x00E				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect	R/W	0x0
15:0	CDSA_U	Upper bits for destination start address.The destination start addresss is up to a 32-bit byte address made of the concatenation of DMA_CDSA_U and DMA_CDAS_L.	R/W	0x0

Table 2–34. Channel Element Number Register (DMA_CEN_CHX)

Base Address = 0xFFFE:D800, Offset = 0x010				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:0	CEN	Channel element number. Number of elements within a frame.	R/W	0x0

Table 2–35. Channel Frame Number Register (DMA_CFN_CHX)

Base Address = 0xFFFE:D800, Offset = 0x012				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:0	CFN	Channel frame number. Number of frames within the block to transfer. The maximum frame number is 65535.	R/W	0x0

Table 2–36. Channel Frame Index Register (DMA_CFI_CHX)

Base Address = 0xFFFE:D800, Offset = 0x014				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:0	CFI	Contains the frame index expressed in bytes, used to compute the adddresses when double index addressing mode used	R/W	0x0

Table 2–37. Channel Element Index Register (DMA_CEI_CHX)

Base Address = 0xFFFE:D800, Offset = 0x016				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:0	CEI	Contains the element index expressed in bytes, used to compute the addresses when single index addressing mode is used.	R/W	0x0

Table 2–38. Channel Progress Counter Register (DMA_CPC_CHX)

Base Address = 0xFFFE:D800, Offset = 0x018				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:0	CPC	This register is used to monitor the progress of DMA transfer.	R/W	0x0

Table 2–39. Channel Destination Address Counter Register (DMA_CDAC_CHX)

Base Address = 0xFFFE:D800, Offset = 0x01A				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	DST_ADDRESS	This register can be used to monitor the progress of a GDMA transfer on channel destination port: It is a snap shot of current destination address generated by channel destination address counter, which is scheduled in channel destination port It is incremented on each accesss, which is made on destination port (S8, S16, or S32)	R	0x0

Table 2–40. DMA Channel Destination Element Index Register (DMA_CDEI_CHX)

Base Address = 0xFFFE:D800, Offset = 0x01C				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	DST_ELEMENT_INDEX	Channel destination element index It contains the channel destination element index, expressed as a signed value in bytes, which is used to compute the addresses, when single/ double index addressing mode is used. When DMA_CCR[10] = 1, then destination_ element_index = GDMA_CDEI When DMA_CCR[10] = 0, then destination_ element_index = GDMA_CSEI	R/W	0x0

Table 2–41. Channel Destination Frame Index Register (DMA_CDFI_CHX)

Base Address = 0xFFFE:D800, Offset = 0x01E				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	DST_FRAME_INDEX	Channel destination frame index It contains the channel destination frame index, expressed as a signed value in bytes, which is used to compute the addresses, when double index addressing mode is used. When DMA_CCR[10] = 1, then destination_ frame_index = GDMA_CDFI When DMA_CCR[10] = 0, then destination_ frame_index = GDMA_CSFI	R/W	0x1

Table 2–42. DMA Color Parameter Register Register (DMA_COLOR_L_CHX)

Base Address = 0xFFFE:D800, Offset = 0x020				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	BLT_FOREGROUND_ COLOR	GDM COLOR parameter This register can be used to provide parameter for DMA constant fill and transparent copy features. It must be configured in big endian format. If DMA_CCR[Constant_Fill_Enable] = 1, then it defines the parameter for constant filling. If data_type = 8-bit, then DMA_COLOR_L[7:0] defines color for constant fill If data_type = 16-bit, then DMA_COLOR_L[15:0] defines color for constant fill. If data_type = 32-bit, then DMA_COLOR_L[15:0] defines parameter[15:0](LSW) for constant filling, DMA_COLOR_U[15:0] defines parameter [31:16](MSW) for constant filling. If DMA_CCR[Tranparent_Copy_Enable] = 1, then it defines the parameter for transparent copy. If data_type = 8-bit, then DMA_COLOR_L[7:0] defines color for transparent copy. If data_type = 16-bit, then DMA_COLOR_L[15:0] defines color for transparent copy. If data_type = 32-bit, then DMA_COLOR_L[15:0] defines parameter[15:0](LSW) for transparent copy, DMA_COLOR_U[15:0] defines parameter [31:16](MSW) for transparent copy.	R/W	0x0

Table 2−43. DMA Color Parameter Register (DMA_COLOR_U_CHX)

Base Address = 0xFFFE:D800, Offset = 0x022				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	BLT_FOREGROUND_ COLOR	GDMA COLOR parameter register This register can be used to provide parameter for DMA constant fill and transparent copy" features. It must be configured in big endian format. This register should only be programmed for data types of 32. If DMA_CCR[Constant_Fill_Enable] = 1, then it defines the parameter for constant filling. If data_type = 32 bit, then DMA_COLOR_L[15:0] defines parameter[15:0](LSW) for constant filling DMA_COLOR_U[15:0] defines parameter [31:16](MSW) for constant filling. If DMA_CCR[Tranparent_Copy_Enable] = 1, then it defines the parameter for transparent copy. If data_type = 32 bit, then DMA_COLOR_L[15:0] defines parameter[15:0](LSW) for transparent copy, DMA_COLOR_U[15:0] defines parameter [31:16](MSW) for transparent copy.	R/W	0x0

Table 2−44. DMA Channel Control Register 2 (DMA_CCR2_CHX)

Base Address = 0xFFFE:D800, Offset = 0x024				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:9	RESERVED1	Undefined	R/W	0x0
8	PACKET_SIZE_CSFI_CDFI	1: Number of element in packet is defined in DMA_CSFI[15:0] 0: Number of element in packet is defined in DMA_CDFI[15:0]	R/W	0xX(Undefined)
7	FRMINDEX_32BITS	0: Frame index is 16 bits 1: Frame index is 32 bits	R/W	0xX(Undefined)
6:3	RESERVED2	Undefined	R/W	0x0

Table 2–44. DMA Channel Control Register 2 (DMA_CCR2_CHX) (Continued)

Base Address = 0xFFFE:D800, Offset = 0x024				
Bit	**Name**	**Function**	**R/W**	**Reset**
2	BS	Block synchronization This bit is used to program the way that a GDMA_request is servived in a synchronized transfer: 1:an entire block is transferred each time, when a GDMA request is made. This frame can be interleaved on the GDMA ports with other channel requests. 0:an element/frame is transferred each time, when a GDMA request is made. The element can be interleaved on the GDMA port with other channel requests. HOWEVER, if DMA_CCR2.bs = 1 AND DMA_CCR.fs = 1, the results are undefined as this is not a valid mode for OMAP 3.2	R/W	0xU
1	TRANSPARENT_COPY_ ENABLE	Transparent copy is enabled 1: Transparent copy operation is enabled. During operation, any source data type that matches the GDMA_COLOR_U/L registers is not written to the destination 0: Transparent copy operation is disabled.	R/W	0xU
0	CONSTANT_FILL_ENABLE	Constant fill operation enable 1:Constant fill operation is enabled. During constant fill operation, it writes destination with GDMA_COLOR_U/L, instead of data from the source. 0:Constant fill operation is disabled. During operation, any source data is written to the destination.	R/W	0xU

Table 2−45. Logical Channel Link Control Register (DMA_CLNK_CTRL_CHX)

Base Address = 0xFFFE:D800, Offset = 0x028				
Bit	**Name**	**Function**	**R/W**	**Reset**
15	ENABLE_LNK	Enable link defines the logical channel is on channel linked queue. 1: The logical channel, defined by NextLCH_ID, is enabled after the current channel finishes transferring 0: No logical channel is chained after the current logical channel	R/W	0x0
14	STOP_LNK	STOP LINK disables the logical channel on the channel linked queue. 1: The logical channel, defined by NextLCH_ID, is disabled and ENABLE_LNK is disabled. 0: No logical channel in chained is disabled.	R/W	0x0
13:4	RESERVED	Undefined	R/W	0xU
3:0	NEXTLCH_ID	NextLCH_ID is used to build logical channel chaining queue. The logical_channel x is enabled after the current logical channel finishes transfer.	R/W	0x0

Table 2−46. DMA Logical Channel Control Register (DMA_LCH_CTRL_CHX)

Base Address = 0xFFFE:D800, Offset = 0x02A				
Bit	**Name**	**Function**	**R/W**	**Reset**
15	LCH_INTERLEAVE_DISABLE	Logical channel interleave disable defines synchronized logical channel interleave mode enable: 0: A synchronized logical channel interleave mode is enabled. The logical channel transfer can be interleaved at the end of each DMA request transfer 1: A synchronized logical channel interleave mode is disabled. The logical channel takes control of the PCH until the entire DMA data has been transferred, regardless of DMA request However, to avoid a LCH transfer (synchronized or not) to be suspended, the priority field of LCH has to be set to 1.	R/W	0x0
14:4	RESERVED	UNDEFINED	R/W	0xU
3:0	LCH_TYPE	LCH_TYPE defines logical channel assignment relationship to physical channel For OMAP GDMA: 0000: LCH-2D dynamically share the 2 PCH-M 0001: LCH-G dynamically share the 2 PCH-M 0010: LCH-P dynamically share the PCM-M (New) 0100: LCH-D uses PCH-D only 1111: LCH_PD uses PCH-P only LCD_CHANNEL (OMAP 3.1 mode), LCH_TYPE is 1111.	R/W	0x0

Table 2–47. Channel Current Destination Element Number (Least Significant Word) Register (DMA_CDDEN_L)

Base Address = 0xFFFE:D800, Offset = 0x034				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	DMA_CDDEN_L	The lower 16-bits of the channel current destination element number	R/W	0xX(Undefined)

Table 2–48. Channel Current Destination Element Number (Most Significant Word) Register (DMA_CDDEN_U)

Base Address = 0xFFFE:D800, Offset = 0x036				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	DMA_CDDEN_U	The upper 16-bits of the channel current destination element number	R/W	0xX(Undefined)

Table 2–49. Extended Source Frame Index (Most Significant Word) Register (DMA_CEXSFI_U)

Base Address = 0xFFFE:D800, Offset = 0x038				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	EXTENDED_SOURCE_ FRAME_INDEX	IF OMAP3.1 mode DMA_CSFI[15:0]: defines the source frame index. ELSE OMAP3.2 mode IF DMA_CCR2.Frmindex_32bit = 0 DMA_CSFI[15:0]: defines the source frame index. ELSE DMA_CEXSFI_U[15:0]: defines the upper 16 bits of the source frame index.	R/W	0xX(Undefined)

Table 2–50. Extended Source Frame Index (Least Significant Word) Register (DMA_CEXSFI_L)

Base Address = 0xFFFE:D800, Offset = 0x03A				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	EXTENDED_SOURCE_ FRAME_INDEX	IF OMAP3.1 mode DMA_CSFI[15:0]: defines the source frame index. ELSE OMAP3.2 mode IF DMA_CCR2.Frmindex_32bit = 0 DMA_CSFI[15:0]: defines the source frame index. ELSE DMA_CEXSFI_L[15:0]: defines the lower 16 bits of the source frame index.	R/W	0xX(Undefined)

Table 2–51. Extended Destination Frame Index (Most Significant Word) Register (DMA_CEXDFI_U)

Base Address = 0xFFFE:D800, Offset = 0x03C				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	EXTENDED_DESTINATION_ FRAME_INDEX	IF OMAP3.1 mode DMA_CDFI[15:0]: defines the destination frame index. ELSE OMAP3.2 mode IF DMA_CCR2.Frmindex_32bit = 0 DMA_CDFI[15:0]: defines the destination frame index. ELSE DMA_CEXDFI_U[15:0]: defines the upper 16 bits of the destination frame index.	R/W	0xX(Undefined)

Table 2–52. Extended Destination Frame Index (Least Significant Word) Register (DMA_CEXDFI_L)

Base Address = 0xFFFE:D800, Offset = 0x03E				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	EXTENDED_DESTINATION_ FRAME_INDEX	IF OMAP3.1 mode DMA_CDFI[15:0]: defines the destination frame index. ELSE OMAP3.2 mode IF DMA_CCR2.Frmindex_32bit = 0 DMA_CDFI[15:0]: defines the destination frame index. ELSE DMA_CEXDFI_L[15:0]: defines the lower 16 bits of the destination frame index.	R/W	0xX(UNdefined)

Table 2–53. LCD Control Register (DMA_LCD_CRTL)

Base Address = 0xFFFE:D800, Offset = 0x300				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:7	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
6	LCD_SOURCE	Memory source for LCD Channel 0: Source is SDRAM 1: Source is IMIF	R/W	0x0
5	BUS_ERROR_IT_COND	Status LCD channel register 0: Bus error interrupt detected 1: No bus error interrupt detected	R	0x0
4	FRAME_2_IT_COND	Status LCD channel register 0: No end frame 2 interrupt detected 1: End frame 2 interrupt detected	R	0x0
3	FRAME_1_IT_COND	Status LCD channel register 0: No end frame 1 interrupt detected 1: End frame 1 interrupt detected	R	0x0

Table 2–53. LCD Control Register (DMA_LCD_CRTL) (Continued)

Base Address = 0xFFFE:D800, Offset = 0x300				
Bit	**Name**	**Function**	**R/W**	**Reset**
2	BUS_ERROR_IT_IE	Bus error interrupt enable 0: Interrupt disable 1: Interrupt enable	R/W	0x0
1	FRAME_IT_IE	END Frame interrupt enable 0: Interrupt disable 1: Interrupt enable	R/W	0x0
0	FRAME_MODE	Kind of frame mode used for LCD transfer 0: One frame buffer, only register for frame 1 is used 1: Two frame buffer, LCD channel reads alternatively Top_frame_1 and2	R/W	0x0

Table 2–54. LCD Top Address for Frame Buffer 1 Lower Bits Register (DMA_LCD_TOP_F1_L)

Base Address = 0xFFFE:D800, Offset = 0x302				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:1	ADD_L	LCD top address for frame buffer 1 lower bits	R/W	0x0
0	ALWAYS0	Always equal to 0	R/W	0x0

Table 2–55. LCD Top Address for Frame Buffer 1 Upper Bits Register (DMA_LCD_TOP_F1_U)

Base Address = 0xFFFE:D800, Offset = 0x304				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:0	ADD_H	LCD top address for frame buffer 1 upper bits	R/W	0x0

Table 2–56. LCD Bottom Address for Frame Buffer 1 Lower Bits Register (DMA_LCD_BOT_F1_L)

Base Address = 0xFFFE:D800, Offset = 0x306				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:1	ADD_L	LCD bottom address for frame buffer 1 lower bits	R/W	0x0
0	ALWAYS0	Always equal to 0	R	0x0

Table 2–57. LCD Bottom Address for Frame Buffer 1 Upper Bits Register (DMA_LCD_BOT_F1_U)

Base Address = 0xFFFE:D800, Offset = 0x308				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:0	ADD_H	LCD bottom address for frame buffer 1 upper bits	R/W	0x0

Table 2–58. LCD Top Address for Frame Buffer 2 Upper Bits Register (DMA_LCD_TOP_F2_L)

Base Address = 0xFFFE:D800, Offset = 0x30A				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:1	ADD_L	LCD top address for frame buffer 2 lower bits	R/W	0x0
0	ALWAYS0	Always equal to 0	R	0x0

Table 2–59. LCD Top Address for Frame Buffer 2 Upper Bits Register (DMA_LCD_TOP_F2_U)

Base Address = 0xFFFE:D800, Offset = 0x30C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect	R/W	0x0
15:0	ADD_H	LCD top address for frame buffer 2 upper bits	R/W	0x0

Table 2–60. LCD Bottom Address for Frame Buffer 2 Lower Bits Register (DMA_LCD_BOT_F2_L)

Base Address = 0xFFFE:D800, Offset = 0x30E				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect	R/W	0x0
15:1	ADD_L	LCD bottom address for frame buffer 2 lower bits	R/W	0x0
0	ALWAYS0	Always equal to 0	R	0x0

Table 2–61. LCD Bottom Address for Frame Buffer 2 Upper Bits Register (DMA_LCD_BOT_F2_U)

Base Address = 0xFFFE:D800, Offset = 0x310				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
15:0	ADD_H	LCD bottom address for frame buffer 2 upper bits	R/W	0x0

Table 2−62. Global Control Register (DMA_GCR)

Base Address = 0xFFFE:D800, Offset = 0x400				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:5	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0
4	PCH_SCHEDULER_ROUND_ROBIN_DISABLE	GDMA physical channel scheduler round robin scheduling disable 0: GDMA physical channel uses round robin scheduling scheme 1: GDMA physical channel uses fixed weighted scheduling scheme (from LCH0 - LCH i) to schedule next available Logical Channel.	R/W	0x0
3	AUTOGATING_ON	Allows the DMA to cut off its clocks according to its activity. 0: Clocks are always ON 1: Autogating enable.	R/W	0x1
2	FREE	DMA reaction to suspend the signal 0: DMA suspends all the current transfers when it receives the suspend signal from the processor 1: DMA continues running when it receives the suspend signal from the processor	R/W	0x0
1:0	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R/W	0x0

Table 2−63.Software Compatible Register (DMA_GSCR)

Base Address = 0xFFFE:D800, Offset = 0x404				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:4	RESERVED	Undefined	R	0x0
3	OMAP_31_MAPPING_DISABLE	OMAP 3.1 mapping disable control 0: GDMA is compatible with OMAP_3.1 GDMA, which maps interrupt lines and LCD channel programming address. Therefore, GDMA logical channel can be configured as OMAP_3.1 compatible channel or OMAP 3.2 compatible channel by configuring GDMA_LCH_CCR[10] 1: GDMA is compatible with OMAP_3.2 GDMA, which maps OMAP_3.2 interrupt lines and LCD channel programming address. Therefore, GDMA logical channel can be configured as OMAP_3.1 compatible channel or OMAP 3.2 compatible channel by configuring GDMA_LCH_CCR[10]	R/W	0x0
2:0	RESERVED	Undefined	R	0x0

Table 2−64. Software Reset Control Register (DMA_GRST)

Base Address = 0xFFFE:D800, Offset = 0x408				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:1	RESERVED	Undefined	R	0x0
0	RESET	GDMA software reset control bit 1: It resets the whole GDMA when software writes a 1 to set this bit. It is reset to 0 by hardware automatically. Therefore, software read to this register is 0.	R/W	0x0

Table 2−65. DMA Hardware Version ID Number Register (DMA_HW_ID)

Base Address = 0xFFFE:D800, Offset = 0x442				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	GDMA_VERSION_ID_NUMBER	GDMA subchip number for each spin of the GDMA, regardless of what feature changes are made	R	0x1

Table 2−66. Pch-P Version ID Register (DMA_PCHP_ID)

Base Address = 0xFFFE:D800, Offset = 0x444				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	GDMA_PCHP_VERSION_ID_NUMBER	Version ID number for peripheral physical channel. GDMA Pch-P version-ID number for OMAP_3.2 GDMA	R	0x0001

Table 2−67. Pch-M0 Version ID Register (DMA_PCHM0_ID)

Base Address = 0xFFFE:D800, Offset = 0x446				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	GDMA_PCHM0_VERSION_ID_NUMBER	Version ID number for memory oriented physical channel. GDMA Pch-M0 version-ID number for OMAP_3.2 GDMA	R	0x0001

Table 2−68. PCH-M1 Version ID Register (DMA_PCHM_1_ID)

Base Address = 0xFFFE:D800, Offset = 0x448				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	GDMA_PCHM1_VERSION_ID_NUMBER	GDMA Pch-M1 version-ID number for OMAP 3.2 GDMA	R	0x0001

Table 2−69. PCH-G Version ID Number (DMA_PCHG_ID)

Base Address = 0xFFFE:D800, Offset = 0x44A				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	GDMA_PCHG_ID	GDMA PCh-G version-ID number for OMAP 3.2 GDMA	R	0x0000

Table 2−70. DMA Pch-D Version ID Number Register (DMA_PCHID_ID)

Base Address = 0xFFFE:D800, Offset = 0x44C				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	GDMA_PCHD_ID	PCh-D version ID number	R	0x0001

Table 2−71. Global DMA Capability Register 0 (DMA_CAPS_0_U)

Base Address = 0xFFFE:D800, Offset = 0x44E				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:4	RESERVED	Undefined	R	0x0
3	CONSTANT_FILL_CAP	Constant fill capability: 0: PCH-G/PCH-M Is not able to do constant fill copy 1: PCH-G/PCH-M is able to do constant fill copy	R	0x1
2	TRANSPARENT_BLT_CAP	Transparent BLT capability: 0: PCH-G/PCH-M is not able to do transparent BLT copy 1: PCH-G/PCH-M is able to do transparent BLT copy	R	0x1
1	OVERLAP_DETECTION_CAP	Overlap detection capability: 0: PCH-G is not able to do overlap detection 1: PCH-G is able to do overlap detection	R	0x0
0	DIRECTIONAL_BLT_CAP	DIRECTIONAL_BLT_CAPABILITY 0: PCH-G is not able to do directional BLT copy 1: PCH-G is able to do directional BLT copy”	R	0x0

Table 2−72. Global DMA Capability Register 0 (DMA_CAPS_0_L)

Base Address = 0xFFFE:D800, Offset = 0x450				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:3	RESERVED	Undefined	R	0x0
2	SUB_BYTE_DESTINATION_CAP	Sub-Byte destination capability: 0: PCH-G is not able to do sub-byte adjust for expansion 1: PCH-G is able to do sub-byte adjust for expansion	R	0x0
1	RESERVED	Undefined	R	0x0
0	ORIGIN_COORDINATE_CAP	Origin coordinate capability: 0: PCH-G is not able to do origin coordinate calculation 1: PCH-G is able to do origin coordinate calculation	R	0x0

Table 2−73. Global DMA Capability Register 1 (DMA_CAPS_1)

Base Address = 0xFFFE:D800, Offset = 0x454				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:2	RESERVED	Undefined	R	0x0
1	ONE_BIT_COLOR_EXPANSION_CAP	1-bit palettized capability 0: PCH-G is not able to do 1-bit color expansion 1: PCH-G is able to do 1-bit color expansion	R	0x0
0	RESERVED	Undefined	R	0x0

Table 2−74. Global DMA Capability Register 2 (DMA_CAPS_2)

Base Address = 0xFFFE:D800, Offset = 0x456				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:9	RESERVED	Undefined	R	0x0
8	SEPARATE_SRC_DST_INDEX_CAP	Separate source/double-index capability 1: Support separate src/dst index for 2-D addressing 0: Not support separate src/dst index for 2-D addressing	R	0x1
7	DST_DOUBLE_INDEX_ADDRESS_CAP	Destination double-index address capability: 1: Support double-index address mode in destination port 0: Not support double-index address mode in destination port	R	0x1
6	DST_SINGLE_INDEX_CAP	Destination single-index address capability 1: Support single-index address mode in destination 0: Not support single-index address mode in destination	R	0x1
5	DST_POST_INCREMENT_CAP	Destination post-increment address capability 1: Support post-increment address mode in destination port 0: Not support post-increment address mode in destination port	R	0x1
4	DST_CONSTANT_CAP	Destination constant address capability 1: Support constant address mode in destination port 0: Not support constant address mode in destination port	R	0x1
3	SOURCE_DOUBLE_INDEX_CAP	Source double-index address capability: 1: Support double-index address mode in source port 0: Not support double-index address mode in source port	R	0x1
2	SOURCE_SINGLE_INDEX_CAP	Source single-index address capability 1: Support single-index address mode in source port 0: Not support single-index address mode in source port	R	0x1

Table 2−74. Global DMA Capability Register 2 (DMA_CAPS_2) (Continued)

Base Address = 0xFFFE:D800, Offset = 0x456				
Bit	**Name**	**Function**	**R/W**	**Reset**
1	SOURCE_POST_INCREMENT_CAP	Source post-increment address capability 1: Support post-increment address mode in source port 0: Not support post-increment address mode in source port	R	0x1
0	SOURCE_CONSTANT_CAP	Source constant address capability 1: Support constant address mode in source port 0: Not support constant address mode in source port	R	0x1

Table 2−75. Global DMA Capability Register 3 (DMA_CAPS_3)

Base Address = 0xFFFE:D800, Offset = 0x458				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:6	RESERVED	Undefined	R	0x0
5	CHANNEL_CHAINING_CAP	Channel channing capability 1: Support logical channel chaining capability 0: Not support logical channel chaining capability	R	0x1
4	LCH_INTERLEAVE_CAP	LCH interleave capability 1: Support logical channel interleave capabiltiy 0: Not support logical channel interleave capability	R	0x1
3	AUTOINIT_REPEAT_CAP	Autoinit-repeat capability 1: Support repeat feature in autoinit mode 0: Not support repeat feature in autoinit mode	R	0x1
2	AUTOINIT_END_PROG_CAP	Autoinit_End_Program capability 1: Support End_Prog feature in autoinit mode 0: Not support End_Prog feature in autoinit mode	R	0x1
1	FRAME_SYNCHRONIZATION_CAP	Frame synchronization capability 1: Support synchronization transfer on frame boundary 0: Not support synchronization transfer on frame boundary	R	0x1
0	ELEMENT_SYNCHRONIZATION_CAP	Element synchronization capability 1: Support synchronization transfer on element boundary 0: Not support synchronization transfer on element bounday	R	0x1

Table 2−76. Global DMA Capability Register 4 (DMA_CAPS_4)

Base Address = 0xFFFE:D800, Offset = 0x45A				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:8	RESERVED	Undefined	R	0x0
7	ACCESS_VIOLATION_ INTERRUPT_CAP	Access violation interrupt capability 1:Support access violation status bit generation 0:Not support access violation status bit generation	R	0x0
6	SYNC_STATUS_CAP	Synchronization status capability 1:Support synchronization transfer status bit generation 0:Not support synchronization transfer status bit generation	R	0x1
5	BLOCK_INTERRUPT_CAP	Block interrupt capability 1: Support block interrupt generation capability 0: Not support block interrupt generation capability	R	0x1
4	LAST_FRAME_INTERRUPT_CAP	Last frame interrupt capability 1:Support last frame interrupt generation capability 0:Not support last frame interrupt generation capability	R	0x1
3	FRAME_INTERRUPT_CAP	Frame interrupt capability 1: Support frame interrupt generation capability 0: Not support frame interrupt generation capability	R	0x1
2	HALF_FRAME_INTERRUPT_CAP	Half frame interrupt capability 1: Support half-frame interrupt generation capability 0: Not support half-frame interrupt generation capability	R	0x1
1	EVENT_DROP_INTERRUPT_CAP	Event drop interrupt generation capability 1: Support event drop interrupt generation capability 0: Not support event drop interrupt generation capability	R	0x1
0	TIMEOUT_INTERRUPT_CAP	Time-out interrupt capability 1: Support time-out interrupt generation capability 0: Not support time-out interrupt generation capability	R	0x1

Table 2−77. DMA Peripheral Physical Channel 0 Status Register (DMA_PCHSR_P_0)

Base Address = 0xFFFE:D800, Offset = 0x460				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:8	RESERVED	Undefined	R	0x0
7:0	ACTIVE_LCH_ID	Active logical channel ID for Pch-P It returns the logical channel ID, which is active in the physical_channel P now. This register can be used to monitor the progress of a GDMA physical channel transfer. It is a snap shot of the current logical channel ID, which is active on the physical channel. FF: GDMA physical channel P is IDLE Others: GDMA logical channel ID	R	0xFF

Table 2−78. DMA Memory Physical Channel 0 Status Register (DMA_PCHSR_M_0)

Base Address = 0xFFFE:D800, Offset = 0x480				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:8	RESERVED	Undefined	R	0x0
7:0	ACTIVE_LCH_ID	Active logical channel ID for Pch-M 0 It returns the logical channel ID, which is active in the physical_channel M0 now. This register can be used to monitor the progress of a GDMA physical channel transfer. It is a snapshot of the current logical channel ID, which is active on the physical channel. ”FF” : GDMA physical channel M0 is IDLE Others: GDMA logical channel ID	R	0xFF

Table 2−79. DMA Memory Physical Channel 1 Status Register (DMA_PCHSR_M_1)

Base Address = 0xFFFE:D800, Offset = 0x482				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:8	RESERVED	Undefined	R	0x0
7:0	ACTIVE_LCH_ID	Active logical channel ID for Pch-M 1 It returns the logical channel ID, which is active in the physical_channel M1 now. This register can be used to monitor the progress of a GDMA physical channel transfer. It is a snap shot of the current logical channel ID, which is active on the physical channel. FF: GDMA physical channel M1 is IDLE Others: GDMA logical channel ID	R	0xFF

Table 2–80. DMA Physical Display Channel 0 Status Register (DMA_PCHSR_D_0)

Base Address = 0xFFFE:D800, Offset = 0x4C0				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:8	RESERVED	Undefined	R	0x0
7:0	ACTIVE_LCH_ID	Active logical channel ID for Pch-D 0 It returns the logical channel ID, which is active in the physical_channel D0 now. This register can be used to monitor the progress of a GDMA physical channel transfer. It is a snap shot of the current logical channel ID, which is active on the physical channel. FF: GDMA physical channel D0 is IDLE Others: GDMA logical channel ID	R	0xFF

2.2.25 Window Tracer Registers (WT)

See ECN5 (Section 2.2.5) and ECN9 (Section 2.2.9) for changes.

Table 2–81 lists the window tracer registers. Table 2–82 through Table 2–101 describe the register bits.

Table 2–81. Window Tracer Registers

Base Address = 0xFFFE:D500				
Register	**Description**	**Bits**	**R/W**	**Offset**
EMIFF_W1_TOPADDR	Top address of the first window of EMIFF	32	R/W	0x00
EMIFF_W1_BOTADDR	Bottom address of the first window of EMIFF	32	R/W	0x04
EMIFF_W2_TOPADDR	Top address of the second window of EMIFF	32	R/W	0x08
EMIFF_W2_BOTADDR	Bottom address of the second window of EMIFF	32	R/W	0x0C
EMIFS_W1_TOPADDR	Top address of the first window of EMIFS	32	R/W	0x10
EMIFS_W1_BOTADDR	Bottom address of the first window of EMIFS	32	R/W	0x14
EMIFS_W2_TOPADDR	Top address of the second window of EMIFS	32	R/W	0x18
EMIFS_W2_BOTADDR	Bottom address of the second window of EMIFS	32	R/W	0x1C
OCPT1_W1_TOPADDR	Top address of the first window of OCPT1	32	R/W	0x20
OCPT1_W1_BOTADDR	Bottom address of the first window of OCPT1	32	R/W	0x24
OCPT1_W2_TOPADDR	Top address of the second window of OCPT1	32	R/W	0x28
OCPT1_W2_BOTADDR	Bottom address of the second window of OCPT1	32	R/W	0x2C
OCPT2_W1_TOPADDR	Top address of the first window of OCPT2	32	R/W	0x30
OCPT2_W1_BOTADDR	Bottom address of the first window of OCPT2	32	R/W	0x34
OCPT2_W2_TOPADDR	Top address of the second window of OCPT2	32	R/W	0x38
OCPT2_W2_BOTADDR	Bottom address of the second window of OCPT2	32	R/W	0x3C
WT_WINENR	Window enable	32	R/W	0x40
WT_LOCK	Window tracer lock	32	R/W	0x44
RES_SPC_ATYPER	Abort handling for reserved space	32	R/W	0x48
RES_SPC_ADDR	Hold address of the transfer that has been aborted	32	R	0x4C

Table 2–82. Top Address of the First Window of EMIFF Register (EMIFF_W1_TOPADDR)

Base Address = 0xFFFE:D500, Offset = 0x00				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–83. Bottom Address of the First Window of EMIFF Register (EMIFF_W1_BOTADDR)

Base Address = 0xFFFE:D500, Offset = 0x04				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–84. Top Address of the Second Window of EMIFF Register (EMIFF_W2_TOPADDR)

Base Address = 0xFFFE:D500, Offset = 0x08				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–85. Bottom Address of the Second Window of EMIFF Register (EMIFF_W2_BOTADDR)

Base Address = 0xFFFE:D500, Offset = 0x0C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–86. Top Address of the First Window of EMIFS Register (EMIFS_W1_TOPADDR)

Base Address = 0xFFFE:D500, Offset = 0x10				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–87. Bottom Address of the First Window of EMIFS Register (EMIFS_W1_BOTADDR)

Base Address = 0xFFFE:D500, Offset = 0x14				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–88. Top Address of the Second Window of EMIFS Register (EMIFS_W2_TOPADDR)

Base Address = 0xFFFE:D500, Offset = 0x18				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–89. Bottom Address of the Second Window of EMIFS Register (EMIFS_W2_BOTADDR)

Base Address = 0xFFFE:D500, Offset = 0x1C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–90. Top Address of the First Window of OCPT1 Register (OCPT1_W1_TOPADDR)

Base Address = 0xFFFE:D500, Offset = 0x20				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–91. Bottom Address of the First Window of OCPT1 Register (OCPT1_W1_BOTADDR)

Base Address = 0xFFFE:D500, Offset = 0x24				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–92. Top Address of the Second Window of OCPT1 Register (OCPT1_W2_TOPADDR)

Base Address = 0xFFFE:D500, Offset = 0x28				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–93. Bottom Address of the Second Window of OCPT1 Register (OCPT1_W2_BOTADDR)

Base Address = 0xFFFE:D500, Offset = 0x2C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–94. Top Address of the First Window of OCPT2 Register (OCPT2_W1_TOPADDR)

Base Address = 0xFFFE:D500, Offset = 0x30				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–95. Bottom Address of the First Window of OCPT2 Register (OCPT2_W1_BOTADDR)

Base Address = 0xFFFE:D500, Offset = 0x34				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–96. Top Address of the Second Window of OCPT2 Register (OCPT2_W2_TOPADDR)

Base Address = 0xFFFE:D500, Offset = 0x38				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–97. Bottom Address of the Second Window of OCPT2 Register (OCPT2_W2_BOTADDR)

Base Address = 0xFFFE:D500, Offset = 0x3C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Top (or bottom) address for window	R/W	0x0

Table 2–98. Window Enable Register of WT (WT_WINENR)

Base Address = 0xFFFE:D500, Offset = 0x40				
Bit	**Name**	**Function**	**R/W**	**Reset**
7	OCPT2_WIN2_EN	Window 2 enable bit of OCPT2 0: Disable 1: Enable	R/W	0x0
6	OCPT2_WIN1_EN	Window 1 enable bit of OCPT2 0: Disable 1: Enable	R/W	0x0
5	OCPT1_WIN2_EN	Window 2 enable bit of OCPT1 0: Disable 1: Enable	R/W	0x0
4	OCPT1_WIN1_EN	Window 1 enable bit of OCPT1 0: Disable 1: Enable	R/W	0x0
3	EMIFF_WIN2_EN	Window 2 enable bit of EMIFF 0: Disable 1: Enable	R/W	0x0
2	EMIFF_WIN1_EN	Window 1 enable bit of EMIFF 0: Disable 1: Enable	R/W	0x0
1	EMIFS_WIN2_EN	Window 2 enable bit of EMIFS 0: Disable 1: Enable	R/W	0x0
0	EMIFS_WIN1_EN	Window 1 enable bit of EMIFS 0: Disable 1: Enable	R/W	0x0

Table 2–99. Window Tracer Lock Register (WT_LOCK)

Base Address = 0xFFFE:D500, Offset = 0x44				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:1	RESERVED	Reserved	R/W	0xX(Undefined)
0	LOCK	Setting 1 to this bit field lock all window tracer registers values. Lock bit can be reset by only a warm or cold reset	R/W	0x0

Table 2–100. Abort Handling for Reserved Space Register (RES_SPC_ATYPER)

Base Address = 0xFFFE:D500, Offset = 0x48				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:1	RESERVED	To ensure software compatibility, reserved bit must be written to 0 and read value must be considered undefined.	R/W	0x0
0	RESVADD_ABORT	0: No abort 1: Reserved address space access abort	R/W	0x0

Register holds the address of the transfer that has been aborted

Table 2–101. Reserved Space Address Register (RES_SPC_ADDR)

Base Address = 0xFFFE:D500, Offset = 0x4C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ABORT_ADDRESS	Register holds the address of the transfer that has been aborted. Valid only when the RES-VADD_ABORT is set.	R	0x10000000

2.2.26 EMIFS Registers

See ECN6 (Section 2.2.6), ECN7 (Section 2.2.7), and ECN11 (Section 2.2.10) for changes.

Table 2–102 lists the EMIFS registers. Table 2–103 through Table 2–119 describe the register bits.

Table 2–102. EMIFS Registers

Base Address = 0xFFFE:CC00				
Register	**Description**	**Bits**	**R/W**	**Offset**
EMIFS_LRUREG	EMIFS LRU priority	32	R/W	0x04
EMIFS_CONFIG	EMIFS configuration	32	R/W	0x0C
FLASH_CFG_0	EMIFS chip-select configuration nCS0	32	R/W	0x10
FLASH_CFG_1	EMIFS chip-select configuration nCS1	32	R/W	0x14
FLASH_CFG_2	EMIFS chip-select configuration nCS2	32	R/W	0x18
FLASH_CFG_3	EMIFS chip-select configuration nCS3	32	R/W	0x1C
EMIFS_TIMEOUT1_REG	EMIFS dynamic priority time-out 1	32	R/W	0x28
EMIFS_TIMEOUT2_REG	EMIFS dynamic priority time-out 2	32	R/W	0x2C
EMIFS_TIMEOUT3_REG	EMIFS dynamic priority time-out 3	32	R/W	0x30
FL_CFG_DYN_WAIT	EMIFS dynamic wait state	32	R/W	0x40
EMIFS_ABORT_ADDR	EMIFS abort address	32	R	0x44
EMIFS_ABORT_TYPE	EMIFS abort type	32	R	0x48
EMIFS_ABORT_TOUT	EMIFS abort time-out	32	R/W	0x4C
FLASH_ACFG_0_I	EMIFS advanced chip configuration nCS0	32	R/W	0x50
FLASH_ACFG_1_I	EMIFS advanced chip configuration nCS1	32	R/W	0x54
FLASH_ACFG_2_I	EMIFS advanced chip configuration nCS2	32	R/W	0x58
FLASH_ACFG_3_I	EMIFS advanced chip configuration nCS3	32	R/W	0x5C

Table 2–103. EMIFS LRU Priority Register (EMIFS_LRUREG)

Base Address = 0xFFFE:CC00, Offset = 0x04				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:12	OCPI	OCPI consecutive access	R/W	0x0000
11:8	DMA	DMA consecutive access	R/W	0x0000
6:4	DSP	Modem consecutive access	R/W	0x000
2:0	MPU	MPU consecutive access	R/W	0x000

Table 2−104. EMIFS Configuration Register (EMIFS_CONFIG)

Base Address = 0xFFFE:CC00, Offset = 0x0C				
Bit	**Name**	**Function**	**R/W**	**Reset**
4	FR	Ready signal. This bit is sampled from the READY pin. This signal is activated by flash devices when output data is not valide.	R	0xX
3	PDE	Global power down enable signal	R/W	0x0
2	PWD_EN	EMIFS power down enable	R/W	0x0
1	BM	MPU boot mode. This bit is sampled at reset from the ARM_BOOT_MODE pin. When set, CS3 is accessible at address 0x00000.	R/W	0xX
0	WP	Write protect. Enables write protection for all flash devices	R/W	0x0

Table 2−105. EMIFS Chip-Select Configuration Register nCS0 (FLASH_CFG_0)

Base Address = 0xFFFE:CC00, Offset = 0x10				
Bit	**Name**	**Function**	**R/W**	**Reset**
31	PGWSTEN	Compatibility note for new PGWST 0: PGWST is specified by bit[15:12] 1: PGWST is specified by bit[30:27]	R/W	0x0
30:27	PGWST	Number of wait states for page mode ROM access within the page. This bit field takes effect only if PGWSTEN bit field is set to 1.	R/W	0x0
26:23	BTWST	Number of wait states for read to write bus transition.	R/W	0x0
22	MAD	Multiplexed address/data bus. Value read from pin.	R/W	0xX
21	FL	0: Address is incremented for the second 16-bit write with 32-bit writes to 16-bit memories. 1: Address is not incremented for the second write with 32-bit writes to 16-bit memories.	R/W	0x0
20	RESERVED	Must be written to 0.	R/W	0x0
18:16	RDMODE	Read mode select	R/W	0x0
15:12	PGWST_WELEN	For read accesses, number of wait states for page mode ROM read within a page. For write accesses, length of /WE pulse duration. This bit field is used as both PGWST/WELEN when PSWSTEN bit field is set to 0. This field specifies only WELEN when the PGWSTEN bit field is set to 1.	R/W	0xF
11:8	WRWST	Number of wait states for write operation.	R/W	0xF

Table 2–105. EMIFS Chip-Select Configuration Register nCS0 (FLASH_CFG_0) (Continued)

Base Address = 0xFFFE:CC00, Offset = 0x10				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:4	RDWST	Number of wait states for asynchronous read operation. Number of inserted clock cycles in Smart3 protocol (value matches the value programmed in Smart3 devices).	R/W	0xF
2	RT	Re-timing control register. 0: The data are not re-timed. 1: The data coming from the external bus are re-timed with CLK.	R/W	0x0
1:0	FCLKDIV	Flash clock divider 00: Divide by 1. 01: Divide by 2. 10: Divide by 4. 11: Divide by 6.	R/W	0x3

SYNC_BOOT = 0 and boot_ti_ace_rom = 0

Table 2–106. EMIFS Chip-Select Configuration Register nCS1 (FLASH_CFG_1)

Base Address = 0xFFFE:CC00, Offset = 0x14				
Bit	**Name**	**Function**	**R/W**	**Reset**
31	PGWSTEN	Compatibility note for new PGWST 0: PGWST is specified by bit[15:12] 1: PGWST is specified by bit[30:27]	R/W	0x0
30:27	PGWST	Number of wait states for page mode ROM access within the page. This bit field takes effect only if PGWSTEN bit field is set to 1.	R/W	0x0
26:23	BTWST	Number of wait states for read to write bus transition.	R/W	0x0
22	MAD	Multiplexed address/data bus. Value read from pin.	R/W	0xX
21	FL	0: Address is incremented for teh second 16-bit write with 32-bit writes to 16-bit memories. 1: Address is not incremented for the second write with 32-bit writes to 16-bit memories.	R/W	0x0
20	RESERVED	Must be written to 1.	R/W	0x1
18:16	RDMODE	Read mode select	R/W	0x0
15:12	PGWST_WELEN	For read accesses, number of wait states for page mode ROM read within a page. For write accesses, length of /WE pulse duration. This bit field is used as both PGWST/WELEN when PSWSTEN bit field is set to 0. This field specifies only WELEN when the PGWSTEN bit field is set to 1.	R/W	0xF

Table 2−106. EMIFS Chip-Select Configuration Register nCS1 (FLASH_CFG_1) (Continued)

Base Address = 0xFFFE:CC00, Offset = 0x14				
Bit	**Name**	**Function**	**R/W**	**Reset**
11:8	WRWST	Number of wait states for write operation.	R/W	0xF
7:4	RDWST	Number of wait states for asynchronous read operation. Number of inserted clock cycles in Smart3 protocol (value matches the value programmed in Smart3 devices).	R/W	0xF
2	RT	Re-timing control register. 0: The data are not re-timed. 1: The data coming from the external bus are re-timed with CLK.	R/W	0x0
1:0	FCLKDIV	Flash clock divider 00: Divide by 1 01: Divide by 2 10: Divide by 4 11: Divide by 6	R/W	0x3

SYNC_BOOT = 0 and boot_ti_ace_rom = 0

Table 2−107. EMIFS Chip-Select Configuration Register nCS2 (FLASH_CFG_2)

Base Address = 0xFFFE:CC00, Offset = 0x18				
Bit	**Name**	**Function**	**R/W**	**Reset**
31	PGWSTEN	Compatibility note for new PGWST 0: PGWST is specified by bit[15:12] 1: PGWST is specified by bit[30:27]	R/W	0x0
30:27	PGWST	Number of wait states for page mode ROM access within the page. This bit field takes effect only if PGWSTEN bit field is set to 1.	R/W	0x0
26:23	BTWST	Number of wait states for read to write bus transition.	R/W	0x0
22	MAD	Multiplexed address/data bus. Value read from pin.	R/W	0xX
21	FL	0: Address is incremented for teh second 16-bit write with 32-bit writes to 16-bit memories. 1: Address is not incremented for the second write with 32-bit writes to 16-bit memories.	R/W	0x0
20	RESERVED	Must be written to 1.	R/W	0x1
18:16	RDMODE	Read mode select	R/W	0x0

Table 2–107. EMIFS Chip-Select Configuration Register nCS2 (FLASH_CFG_2) (Continued)

Base Address = 0xFFFE:CC00, Offset = 0x18				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:12	PGWST_WELEN	For read accesses, number of wait states for page mode ROM read within a page. For write accesses, length of /WE pulse duration. This bit field is used as both PGWST/WELEN when PSWSTEN bit field is set to 0. This field specifies only WELEN when the PGWSTEN bit field is set to 1.	R/W	0xF
11:8	WRWST	Number of wait states for write operation.	R/W	0xF
7:4	RDWST	Number of wait states for asynchronous read operation. Number of inserted clock cycles in Smart3 protocol (value matches the value programmed in Smart3 devices).	R/W	0xF
2	RT	Re-timing control register. 0: The data are not re-timed. 1: The data coming from the external bus are re-timed with CLK.	R/W	0x0
1:0	FCLKDIV	Flash clock divider 00: Divide by 1 01: Divide by 2 10: Divide by 4 11: Divide by 6	R/W	0x3

SYNC_BOOT = 0 and boot_ti_ace_rom = 0

Table 2–108. EMIFS Chip-Select Configuration Register nCS3 (FLASH_CFG_3)

Base Address = 0xFFFE:CC00, Offset = 0x1C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31	PGWSTEN	Compatibility note for new PGWST 0: PGWST is specified by bit[15:12] 1: PGWST is specified by bit[30:27]	R/W	0x0
30:27	PGWST	Number of wait states for page mode ROM access within the page. This bit field takes effect only if PGWSTEN bit field is set to 1.	R/W	0x0
26:23	BTWST	Number of wait states for read to write bus transition.	R/W	0x0
22	MAD	Multiplexed address/data bus. Value read from pin.	R/W	0xX
21	FL	0: Address is incremented for teh second 16-bit write with 32-bit writes to 16-bit memories. 1: Address is not incremented for the second write with 32-bit writes to 16-bit memories	R/W	0x0
20	RESERVED	Must be written to 0.	R/W	0x0

Table 2−108. EMIFS Chip-Select Configuration Register nCS3 (FLASH_CFG_3) (Continued)

Base Address = 0xFFFE:CC00, Offset = 0x1C				
Bit	**Name**	**Function**	**R/W**	**Reset**
18:16	RDMODE	Read mode select	R/W	0x0
15:12	PGWST_WELEN	For read accesses, number of wait states for page mode ROM read within a page. For write accesses, length of /WE pulse duration. This bit field is used as both PGWST/WELEN when PSWSTEN bit field is set to 0. This field specifies only WELEN when the PGWSTEN bit field is set to 1.	R/W	0xF
11:8	WRWST	Number of wait states for write operation.	R/W	0xF
7:4	RDWST	Number of wait states for asynchronous read operation. Number of inserted clock cycles in Smart3 protocol (value matches the value programmed in Smart3 devices).	R/W	0xF
2	RT	Re-timing control register. 0: The data are not re-timed. 1: The data coming from the external bus are re-timed with CLK.	R/W	0x0
1:0	FCLKDIV	Flash clock divider 00: Divide by 1 01: Divide by 2 10: Divide by 4 11: Divide by 6	R/W	0x3

SYNC_BOOT = 0 and boot_ti_ace_rom = 0

Table 2−109. EMIFS Dynamic Priority time-out 1 (EMIFS_TIMEOUT1_REG)

Base Address = 0xFFFE:CC00, Offset = 0x28				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:0	DMA	Store the number of clock cycles before DMA requests are made high priority in dynamic priority arbitration.	R/W	0x0

Table 2−110. EMIFS Dynamic Priority time-out 2 (EMIFS_TIMEOUT2_REG)

Base Address = 0xFFFE:CC00, Offset = 0x2C				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:0	DSP	Store the number of clock cycles before DSP requests are made high priority in dynamic priority arbitration.	R/W	0x0

Table 2−111. EMIFS Dynamic Priority time-out 3 (EMIFS_TIMEOUT3_REG)

Base Address = 0xFFFE:CC00, Offset = 0x30				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:0	OCPI	Store the number of clock cycles before OCPI requests are made high priority in dynamic priority arbitration.	R/W	0x0

Table 2−112. EMIFS Dynamic Wait State Register (FL_CFG_DYN_WAIT)

Base Address = 0xFFFE:CC00, Offset = 0x40				
Bit	**Name**	**Function**	**R/W**	**Reset**
11:8	WRRDYMASK	0: Considers the ready to extend the write access timing only when full handshaking mode is enabled. 1: Masks the ready signal during write operation.	R/W	0x0
7:4	HANDSHAKE_ENABLES	Setting 1 to this bit field disables the full handshaking mode to the EMIFS for CS0-CS3	R/W	0x0
3:0	WAIT_STATE_ENABLES	Setting 1 to this bit field enables the dynamic wait configuration mode for CS0-CS3	R/W	0x0

Table 2−113. EMIFS Abort Address Register (EMIFS_ABORT_ADDR)

Base Address = 0xFFFE:CC00, Offset = 0x44				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ABORT_ADDR	Address of the abort access	R	0x0

Table 2−114. EMIFS Abort Type Register (EMIFS_ABORT_TYPE)

Base Address = 0xFFFE:CC00, Offset = 0x48				
Bit	**Name**	**Function**	**R/W**	**Reset**
4	time-out_ERR	Time-out error 0: No error 1: Error	R	0x0
3	SECURE_ERR	Restricted access mode error 0: No error 1: Error	R	0x0
2:1	HOST_ID	00: MPU 01: Modem 10: DMA 11: OCPI	R	0x0
0	ABORT	0: No abort 1: Abort	R	0x0

Table 2−115. EMIFS Abort time-out Register (EMIFS_ABORT_TOUT)

Base Address = 0xFFFE:CC00, Offset = 0x4C				
Bit	**Name**	**Function**	**R/W**	**Reset**
8	time-out_EN	time-out enable 0: Disable 1: Enable	R/W	0x1
7:0	time-out_VALUE	Time-out value	R/W	0xFF

Table 2–116. EMIFS Advanced Chip Configuration Register nCS0 (FLASH_ACFG_0_I)

Base Address = 0xFFFE:CC00, Offset = 0x50				
Bit	**Name**	**Function**	**R/W**	**Reset**
11	READY_CONFIG	0: Ready is monitored 1 clock cycle ahead of the data phase. 1: Ready monitored in same cycle than the data phase.	R/W	0x0
10	CLKMASK	0: Flash clock is toggling during write operations for RDMODE 4 and 5. 1: Flash clock is driven low during write operations for RDMODE 4 and 5.	R/W	0x0
9	BTMODE	Mode selection for BTWST 0: Bus turn around for RD -> any transaction 1: Bus turn around for RD -> any transaction plus WR -> WR and WR -> RD of the same chip select.	R/W	0x0
8	ADVHOLD	Hold cycle for address valid signal in async and sync protocol 1 and 2 0: Hold cycle (adv length is 1 flash clock cycle) 1: 1 flash clock cycle hold (adv length is 2 flash clock cycles)	R/W	0x0
7:4	OEHOLD	Number of cycles from OE high to CS high	R/W	0x0
3:0	OESETUP	Number of cycles inserted from CS low to OE low.	R/W	0x0

Note: Mux_device = 0

Table 2–117. EMIFS Advanced Chip Configuration Register nCS1 (FLASH_ACFG_1_I)

Base Address = 0xFFFE:CC00, Offset = 0x54				
Bit	**Name**	**Function**	**R/W**	**Reset**
11	READY_CONFIG	0: Ready is monitored 1 clock cycle ahead of the data phase. 1: Ready monitored in same cycle than the data phase.	R/W	0x0
10	CLKMASK	0: Flash clock is toggling during write operations for RDMODE 4 and 5. 1: Flash clock is driven low during write operations for RDMODE 4 and 5.	R/W	0x0
9	BTMODE	Mode selection for BTWST 0: Bus turn around for RD -> any transaction 1: Bus turn around for RD -> any transaction plus WR -> WR and WR -> RD of the same chip select.	R/W	0x0
8	ADVHOLD	Hold cycle for address valid signal in async and sync protocol 1 and 2 0: Hold cycle (adv length is 1 flash clock cycle) 1: 1 flash clock cycle hold (adv length is 2 flash clock cycles)	R/W	0x0
7:4	OEHOLD	Number of cycles from OE high to CS high	R/W	0x0
3:0	OESETUP	Number of cycles inserted from CS low to OE low	R/W	0x0

Note: Mux_device = 0

Table 2−118. EMIFS Advanced Chip Configuration Register nCS2 (FLASH_ACFG_2_I)

Base Address = 0xFFFE:CC00, Offset = 0x58				
Bit	**Name**	**Function**	**R/W**	**Reset**
11	READY_CONFIG	0: Ready is monitored 1 clock cycle ahead of the data phase. 1: Ready monitored in same cycle than the data phase.	R/W	0x0
10	CLKMASK	0: Flash clock is toggling during write operations for RDMODE 4 and 5. 1: Flash clock is driven low during write operations for RDMODE 4 and 5.	R/W	0x0
9	BTMODE	Mode selection for BTWST 0: Bus turn around for RD -> any transaction 1: Bus turn around for RD -> any transaction plus WR -> WR and WR -> RD of the same chip select.	R/W	0x0
8	ADVHOLD	Hold cycle for address valid signal in async and sync protocol 1 and 2 0: Hold cycle (adv length is 1 flash clock cycle) 1: 1 flash clock cycle hold (adv length is 2 flash clock cycles)	R/W	0x0
7:4	OEHOLD	Number of cycles from OE high to CS high	R/W	0x0
3:0	OESETUP	Number of cycles inserted from CS low to OE low	R/W	0x0

Note: Mux_device = 0

Table 2−119. EMIFS Advanced Chip Configuration Register nCS3 (FLASH_ACFG_3_I)

Base Address = 0xFFFE:CC00, Offset = 0x5C				
Bit	**Name**	**Function**	**R/W**	**Reset**
11	READY_CONFIG	0: Ready is monitored 1 clock cycle ahead of the data phase. 1: Ready monitored in same cycle than the data phase.	R/W	0x0
10	CLKMASK	0: Flash clock is toggling during write operations for RDMODE 4 and 5. 1: Flash clock is driven low during write operations for RDMODE 4 and 5.	R/W	0x0
9	BTMODE	Mode selection for BTWST 0: Bus turn around for RD -> any transaction 1: Bus turn around for RD -> any transaction plus WR -> WR and WR -> RD of the same chip select.	R/W	0x0
8	ADVHOLD	Hold cycle for address valid signal in async and sync protocol 1 and 2 0: Hold cycle (adv length is 1 flash clock cycle) 1: 1 flash clock cycle hold (adv length is 2 flash clock cycles)	R/W	0x0
7:4	OEHOLD	Number of cycles from OE high to CS high	R/W	0x0
3:0	OESETUP	Number of cycles inserted from CS low to OE low	R/W	0x0

Note: Mux_device = 0

2.2.27 EMIFF Registers

See ECN13 (Section 2.2.11) and ECN14 (Section 2.2.12) for changes.

Table 2–120 lists the EMIFF registers. Table 2–121 through Table 2–142 describe the register bits.

Table 2–120. EMIFF Registers

Base Address = 0xFFFE:CC00				
Register	**Description**	**Bits**	**R/W**	**Offset**
EMIFF_PRIORITY_REG	EMIFF priority	32	R/W	0x08
EMIFF_SDRAM_CONFIG	EMIFF SDRAM configuration	32	R/W	0x20
EMIFF_MRS	EMIFF SDRAM MRS	32	R/W	0x24
EMIFF_CONFIG_REG2	EMIFF SDRAM configuration 2	32	R/W	0x3C
EMIFF_DOUBLER_EN	Enable and disable TC doubler feature	32	R/W	0x60
DLL_WRT_CONTROL	DLL generating delayed clock control for DDR write operations	32	R/W	0x64
DLL_WRT_STATUS	DLL generating delayed clock status for DDR write operations	32	R	0x68
EMIFF_MRS_NEW	EMIFF SDRAM MRS	32	R/W	0x70
EMIFF_EMRS0	Extended mode for DDR SDRAM memories	32	R/W	0x74
EMIFF_EMRS1	Extended mode for low power SDRAM memories only	32	R/W	0x78
SDRAM_OPERATION_ REG	Configuration parameters for EMIFF operation	32	R	0x80
SDRAM_MANUAL_ CMD_REG	EMIFF manual command	32	R	0x84
EMIFF_TIMEOUT1	EMIFF time-out 1 for simultaneous access	32	R/W	0x8C
EMIFF_TIMEOUT2	EMIFF time-out 2 for simultaneous access	32	R/W	0x90
EMIFF_TIMEOUT3	EMIFF time-out 3 for simultaneous access	32	R/W	0x94
EMIFF_ABORT_ADDRESS	Address of aborted transfer	32	R	0x98
EMIFF_ABORT_TYPE	Type of aborted transfer	32	R	0x9C
DLL_LRD_STATUS	DLL generating delayed DQS status for lower byte of DDR read data	32	R	0xBC
DLL_URD_CONTROL	DLL generating delayed DQS control for upper byte of DDR read data	32	R/W	0xC0
DLL_URD_STATUS	DLL generating delayed DQS status for upper byte of DDR read data	32	R	0xC4
EMIFF_EMRS2	Reserve extended mode	32	R	0xC8
DLL_LRD_CONTROL	DLL generating delayed DQS control for lower byte of DDR read data	32	R/W	0xCC

Table 2−121. EMIFF Priority Register (EMIFF_PRIORITY_REG)

Base Address = 0xFFFE:CC00, Offset = 0x08				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R	0x0
15:12	L3_OCP	L3 OCP consecutive accesses	R/W	0x0
11:8	DMA	DMA consecutive access	R/W	0x0
7	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R	0x0
6:4	DSP	Modem consecutive access	R/W	0x0
3	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R	0x0
2:0	MPU	MPU consecutive access	R/W	0x0

Table 2−122. EMIFF SDRAM Configuration Register (EMIFF_SDRAM_CONFIG)

Base Address = 0xFFFE:CC00, Offset = 0x20				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:30	RESERVED	Reading this bit gives an undefined value and writing to it has no effect.	R	0x0
29:28	LG_SDRAM_TYPE	Used to define large SDRAM type(> 256 Mb)	R/W	0x0
27	CLK	Disable SDRAM clock. 0: Clock enable 1: Clock disable	R/W	0x0
26	PWD	Power down enable. Put the SDRAM interface into power down enable.	R/W	0x0
25:24	SDRAMFREQ	SDRAM frequency range. To control the latency of SDRAM regarding to the clock organization. 00: SDF0 01: SDF1 10: SDF2 11: SDF3	R/W	0x0
23:8	ARCV	Autorefresh counter value	R/W	0x6188
7:4	SDRAMTYPE	Set the SDRAM internal organization (default: 8-bit bus/16Mb/2 banks)	R/W	0x0
3:2	ARE	Autorefresh enable 00: Autorefresh enable 01: Auto-refresh enable (one command every 14.7us) 10: Autorefresh by burst of 4 commands 11: Autorefresh by burst of 8 commands	R/W	0X0
1	RESERVED	Reserved. Should always be written with 1	R	0x1
0	SLRF	SDRAM put in self-refresh mode when set to 1	R/W	0x0

Table 2–123. EMIFF SDRAM MRS Register (EMIFF_MRS)

Base Address = 0xFFFE:CC00, Offset = 0x24				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:10	RESERVED	Reserved. Must be 00	R/W	0x0
9	WBST	Write burst must be 0 (Burst write same as Burst read)	R/W	0x0
8:7	RESERVED	Reserved. Must be 00	R/W	0x0
6:4	CASL	CAS Latency 001: Reserved 010: CAS Latency = 2 011: CAS Latency = 3 (reset)	R/W	0x3
3	SORI	Serial = 0/Interleave =1 (must be serial)	R/W	0x0
2:0	PGBL	Page burst length. 111 burst full page length. The burst length must be set to 111 (full page burst)	R/W	0x7

Table 2–124. EMIFF SDRAM Configuration Register (EMIFF_CONFIG_REG2)

Base Address = 0xFFFE:CC00, Offset = 0x3C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:4	RESERVED	Reserved. Must be 0	R	0x0
3	NEW_SYS_FREQ	When set to 1, indicates system frequency is greater than 100 MHz. When set, this bit along with SDRAMFREQ = EMIFF_SDRAM_CONFIG[25:24] selects the optimized AC table. When 0, SDRAMFREQ = EMIFF_SDRAM_CONFIG[25:24] selects the OMAP3.2 AC Table.	R/W	0x0
2	SD_AUTO_CLK	Allows controller to suspend internal clocks when idle. The clocks are automatically re-enabled when there is a auto-refresh or host-refresh. 0: Disable internal clocks when idle. 1: Always enable internal clocks.	R/W	0x0
1	SLFR_RESET	Place the SDRAM into self-refresh when there is a warm reset.	R/W	0x1
0	SLFR_STBY	Place the SDRAM into self-refresh when there is standby (chip idle) request from clkrst.	R/W	0x1

Enable and disable TC doubler feature.

Table 2–125. EMIFF TC Doubler Register (EMIFF_DOUBLER_EN)

Base Address = 0xFFFE:CC00, Offset = 0x60				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:1	RESERVED	Reserved	R/W	0xX(Undefined)
0	DOUBLER_EN	Setting 1 to this bit field enables the TC Doubler feature.	R/W	0x0

This register controls the DLL generating delayed clock for DDR write operations.

Table 2−126. DLL Write Operations Register (DLL_WRT_CONTROL)

Base Address = 0xFFFE:CC00, Offset = 0x64				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:26	RESERVED	Reserved. Must be 0.	R	0x0
25:20	READ_OFFSET	6 bit fine delay. Range -16...+15. One step represent a 34ps +- 15 ps delay element.	R/W	0x0
19:16	RESERVED	Reserved. Must be 0.	R	0x0
15:8	DELAY	8 bit delay to adjust DLL. Used when DLL is disabled. range 0..224. One step represents 34ps+- 15ps delay.	R/W	0x0
7:4	RESERVED	Reserved. Must be 0	R	0x0
3	LOAD_DLL	Allows loading a delay field value into the DLL. 0: No action. 1: No action. 0>1: Load occurs on a low to high transition.	R/W	0x0
2	DLL_PHASE	Nominal DLL delay selection. This bit has no effect if DLL is disabled. 0: Delay = 20 % of clk period. 1: Delay = 25 % of clk period.	R/W	0x0
1	ENABLE_DLL	Enable DLL. 0: DLL disabled 1: DLL enabled	R/W	0x0
0	RESERVED	Reserved. Must be 0	R	0x0

This register contains the status information for the DLL generating the delayed clock for DDR write operations.

Table 2−127. DLL Write Status Register (DLL_WRT_STATUS)

Base Address = 0xFFFE:CC00, Offset = 0x68				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reserved. Must be 0	R	0x0
15:8	DLL_COUNT	Current DLL counter value	R	0x0
7:3	RESERVED	Reserved. Must be 0	R	0x0
2	LOCK	DLL lock status. 0: DLL is not locked. 1: DLL is properly locked.	R	0x0
1	UDF	DLL underflow status. If 1, DLL has underflowed.	R	0x0
0	OVF	DLL overflow. If 1 DLL has overflowed.	R	0x0

Table 2−128. EMIF Fast Interface SDRAM MRS Register (EMIFF_MRS_NEW)

Base Address = 0xFFFE:CC00, Offset = 0x70				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:10	RESERVED	Reserved. Must be 00	R/W	0x0
9	WBST	Write burst must be 0 (burst write same as burst read)	R/W	0x0
8:7	RESERVED	Reserved. Must be 00	R/W	0x0
6:4	CASL	CAS latency 001: Reserved 010: CAS Latency = 2 011: CAS Latency = 3 (reset)	R/W	0x3
3	SORI	Serial = 0/Interleave =1 (must be serial)	R/W	0x0
2:0	PGBL	Page burst length. 111 burst full page length. The Burst length must be set to 111 (full page) for SDRAM and to 011 (Burst of 8) for DDR SDRAMs	R/W	0x7

Extended mode register for DDR SDRAM memories.

Table 2−129. Extended Mode Register 0 (EMIFF_EMRS0)

Base Address = 0xFFFE:CC00, Offset = 0x74				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:3	RESERVED	Reserved. Must be 0	R	0x0
2	QFC	QFC enable bit. 0: Disabled 1: Enabled	R/W	0x0
1	DS	Drive strength bit. 0: Normal 1: Reduced	R/W	0x0
0	DLL	DLL enable bit. 0: Enabled 1: Disabled	R/W	0x0

To be used with DDR SDRAM memories only.

Extended mode register for low-power SDRAM memories only.

Table 2−130. Extended Mode Register 1 (EMIFF_EMRS1)

Base Address = 0xFFFE:CC00, Offset = 0x78				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:5	RESERVED	Reserved. Must be 0	R	0x0
4:3	TCSR	Temperature controlled self refresh. 00: 70°C max temperature. 01: 45°C max temperature. 10: 15°C max temperature. 11: 85°C max temperature.	R/W	0x00
2:0	PASR	Partial array self-refresh. 000: All banks 001: 1/2 banks 010: 1/4 banks 011: Reserved 1XX: Reserved	R/W	0x0

This register contains the configuration parameters for EMIFF operation. (memory type, operating mode)

Table 2−131. SDRAM Operation Register (SDRAM_OPERATION_REG)

Base Address = 0xFFFE:CC00, Offset = 0x80				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:25	BANK4_TIMEOUT	Time-out value for BANK4 in clk cycles. Used with POM0 operating mode.	R/W	0x0
24:18	BANK3_TIMEOUT	Time-out value for bank 3 in clk cycles. Used in POM0 operating mode.	R/W	0x0
17:11	BANK2_TIMEOUT	Time-out value for bank 2 in clk cycles. Used in POM0 operating mode.	R/W	0x0
10:4	BANK1_TIMEOUT	Time-out value for bank 1 in clk cycles. Used in POM0 operating mode.	R/W	0x0
3:2	OPERATING_MODE	EMIFF operating mode. 00: Low power mode. 01: High bandwidth mode. 10: Programmable operating mode 0 (POM0). 11: Reserved	R/W	0x1
1	SDRAM_POWER	SDRAM device power. 0: Regular 1: Mobile (low power)	R/W	0x0
0	SDRAM_TYPE	SDRAM device type. 0: SDRAM 1: DDR SDRAM	R/W	0x0

Emiff manual command register. A write to this register issues commands manually to the sdram.

Table 2−132. EMIFF Manual Command Register (SDRAM_MANUAL_CMD_REG)

Base Address = 0xFFFE:CC00, Offset = 0x84				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:4	RESERVED	Reserved. Must be 0	R	0x0
3:0	SDRAM_MANUAL_CMD	The manual command to be issued to the SDRAM. 0000: NOP command 0001: Precharge command 0010: Autorefresh command 0011: Enter deep sleep command. 0100: Exit deep sleep command. 0111: Set CKE high command 1000: Set CKE low command 1001 to 1111: Reserved	R/W	0x0

EMIFF TIMEOUT 1 Register for simultaneous access

Table 2−133. EMIFF TIMEOUT Register 1 (EMIFF_TIMEOUT1)

Base Address = 0xFFFE:CC00, Offset = 0x8C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:8	RESERVED	Reserved. Must be 00	R/W	0x0
7:0	DMA	DMA time-out counter value	R/W	0x0

TIMEOUT2 for simultaneous access.

Table 2−134. EMIFF TIMEOUT Register 2 (EMIFF_TIMEOUT2)

Base Address = 0xFFFE:CC00, Offset = 0x90				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:24	RESERVED	Reserved. Must be 00	R	0x0
23:16	DSP	DSP time-out counter value	R/W	0x0
15:8	RESERVED	Reserved. Must be 00	R	0x00
7:0	LCD	LCD time-out counter value	R/W	0x0

TIMEOUT3 for simultaneous access

Table 2−135. EMIFF TIMEOUT Register 3 (EMIFF_TIMEOUT3)

Base Address = 0xFFFE:CC00, Offset = 0x94				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:8	RESERVED	Reserved. Reading this bit gives an undefined value and writing to it has no effect.	R	0x0
7:0	L3_OCPI	L3_OCPI time-out counter value	R/W	0x0

This register contains the address of an aborted transfer.

Table 2−136. EMIFF Aborted Transfer Address Register (EMIFF_ABORT_ADDRESS)

Base Address = 0xFFFE:CC00, Offset = 0x98				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ABORT_ADDRESS	Address of the transfer aborted.	R	0x0

This register contains the type of abort for an aborted transfer.

Table 2−137. EMIFF Aborted Type Register (EMIFF_ABORT_TYPE)

Base Address = 0xFFFE:CC00, Offset = 0x9C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:3	RESERVED	Reserved. Must be 0	R	0x0
2:1	HOSTID	Host Id whose transfer was aborted. 00: MPU 01: Modem 10: DMA 11: OCPI	R	0x0
0	ABORT_FLAG	When 1, indicates an abort has occured. Cleared on a read.	R	0x0

This register contains the status information for the DLL generating the delayed DQS for lower byte of DDR read data.

Table 2−138. DLL Lower Byte Status Register (DLL_LRD_STATUS)

Base Address = 0xFFFE:CC00, Offset = 0xBC				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reserved. Must be 0	R	0x0
15:8	DLL_COUNT	Current DLL counter value	R	0x0
7:3	RESERVED	Reserved. Must be 0	R	0x0
2	LOCK	DLL Lock status 0: DLL is not locked 1: DLL is properly locked	R	0x0
1	UDF	DLL underflow status. If 1, DLL has underflowed.	R	0x0
0	OVF	DLL overflow. If 1 DLL has overflowed.	R	0x0

This register controls the DLL generating delayed DQS for the upper byte of DDR read data.

Table 2−139. DLL Upper Byte Control Register (DLL_URD_CONTROL)

Base Address = 0xFFFE:CC00, Offset = 0xC0				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:26	RESERVED	Reserved. Must be 0	R	0x0
25:20	READ_OFFSET	6 bit fine delay. Range -16...+15. One step represent a 34ps +- 15 ps delay element.	R/W	0x0

Table 2–139. DLL Upper Byte Control Register (DLL_URD_CONTROL) (Continued)

Base Address = 0xFFFE:CC00, Offset = 0xC0				
Bit	**Name**	**Function**	**R/W**	**Reset**
19:16	RESERVED	Reserved. Must be 0	R	0x0
15:8	DELAY	8 bit delay to adjust DLL. Used when DLL is disabled. range 0..224. One step represents 34ps+- 15ps delay.	R/W	0x0
7:4	RESERVED	Reserved. Must be 0	R	0x0
3	LOAD_DLL	Allows loading a delay field value into the DLL. 0: No action 1: No action 0>1: Load occurs on a low to high transition.	R/W	0x0
2	DLL_PHASE	Nominal DLL delay selection. This bit has no effect if DLL is disabled. 0: Delay = 20 % of clk period 1: Delay = 25 % of clk period	R/W	0x0
1	ENABLE_DLL	Enable DLL. 0: DLL disabled 1: DLL enabled	R/W	0x0
0	RESERVED	Reserved. Must be 0	R	0x0

This register contains the status information for the DLL generating the delayed DQS for upper byte of DDR read data.

Table 2–140. DLL Upper Byte Status Register (DLL_URD_STATUS)

Base Address = 0xFFFE:CC00, Offset = 0xC4				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	RESERVED	Reserved. Must be 0	R	0x0
15:8	DLL_COUNT	Current DLL counter value	R	0x0
7:3	RESERVED	Reserved. Must be 0	R	0x0
2	LOCK	DLL Lock status. 0: DLL is not locked. 1: DLL is properly locked.	R	0x0
1	UDF	DLL underflow status. If 1, DLL has underflowed.	R	0x0
0	OVF	DLL overflow. If 1 DLL has overflowed.	R	0x0

Reserve extended mode register. Not to be used with OMAP3.2

Table 2–141. EMIFF Reserved Extended Mode Register (EMIFF_EMRS2)

Base Address = 0xFFFE:CC00, Offset = 0xC8				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	RESERVED	Reserved. Must be 0	R	0x0

This register controls the DLL generating delayed DQS for the lower byte of DDR read data.

Table 2−142. DLL Lower Byte Control Register (DLL_LRD_CONTROL)

Base Address = 0xFFFE:CC00, Offset = 0xCC				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:26	RESERVED	Reserved. Must be 0	R	0x0
25:20	READ_OFFSET	6 bit fine delay. Range -16...+15. One step represent a 34ps +- 15 ps delay element.	R/W	0x0
19:16	RESERVED	Reserved. Must be 0	R	0x0
15:8	DELAY	8 bit delay to adjust DLL. Used when DLL is disabled. range 0..224. One step represents 34ps+- 15ps delay.	R/W	0x0
7:4	RESERVED	Reserved. Must be 0	R	0x0
3	LOAD_DLL	Allows loading a delay field value into the DLL. 0: No action 1: No action 0>1: Load occurs on a low to high transition.	R/W	0x0
2	DLL_PHASE	Nominal DLL delay selection. This bit has no effect if DLL is disabled. 0: Delay = 20% of clk period 1: Delay = 25% of clk period	R/W	0x0
1	ENABLE_DLL	Enable DLL. 0: DLL disabled 1: DLL enabled	R/W	0x0
0	RESERVED	Reserved. Must be 0	R	0x0

2.2.28 LCD Controller Registers

See ECN27 (Section 2.2.20), BTS12 (Section 2.2.22), and BTS13 (Section 2.2.23) for changes.

Table 2–143 lists the LCD controller registers. Table 2–144 through Table 2–151 describe the register bits.

Table 2–143. LCD Controller Registers

Base Address = 0xFFFE:C000				
Register	**Description**	**Bits**	**R/W**	**Offset**
LCD_CTRL_REG	LCD control	32	R/W	0x00
LCD_TIMING_0	LCD timing 0	32	R/W	0x04
LCD_TIMING_1	LCD timing 1	32	R/W	0x08
LCD_TIMING_2	LCD timing 2	32	R/W	0x0C
LCD_STATUS	LCD status	32	R	0x10
LCD_SUB_PANEL	LCD sub panel display	32	R/W	0x14
LINE_INTERRUPT_ REGISTER	Line interrupt	32	R/W	0x18
DISPLAY_STATUS_ REGISTER	Line status	32	R	0x1C

Table 2–144. LCD Control Register (LCD_CTRL_REG)

Base Address = 0xFFFE:C000, Offset = 0x00				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:27	RESERVED	Reserved	R/W	0x00
26	ALIGNMENT_BIT	0: Misalignment between data and pixel clock 1: Alignment between data and pixel clock	R/W	0x0
25	DPS_EN	Dynamic power saving mode. Enables autoclock gating if enabled. 0: Autoclock gating OFF 1: Autoclock gating ON	R/W	0x0
24	STN_565	0: 12BPP(565) mode is off 1: On (16-bit data in frame buffer, but only 12 bits are dithered and sent out)	R/W	0x0
23	TFT_MAP	0: Output pixel data for 1,2,4,8 and 12BPP modes is right aligned on pixel data [11:0] 1: Output pixel data for 1,2,4,8 and 12BPP modes is converted to 5,6,5 format and uses pins [15:0]	R/W	0x0
22	NM	Nibble mode 0: Nibble mode is disabled 1: Nibble mode is enabled	R/W	0x0

Table 2−144. LCD Control Register (LCD_CTRL_REG) (Continued)

Base Address = 0xFFFE:C000, Offset = 0x00				
Bit	**Name**	**Function**	**R/W**	**Reset**
21:20	PLM	Palette loading mode 00: Palette and data loading, reset value 01: Palette loading 10: Data loading	R/W	0x0
19:12	FDD	FIFO DMA request delay. Value (0-255) used to specify the number of lcd clock. Programming FDD = 00h disables this function.	R/W	0x0
11	PXL_GATED	0: Pixel clock always toggles in TFT mode 1: Pixel clock only toggles when there is valid data to display in TFT mode	R/W	0x0
10	LINE_INT_CLR_SEL	0: TIPB writes 0 to clear line interrupt status register 1: Line interrupt is reset at the end of the programmed line	R/W	0x0
9	M8B	0: In mono 8 bit mode, LCD_pixel_o[3:0] is used to output four pixel values to the panel each pixel clock 1: lcd_pixel_o[7:0] is used to output eight pixel values to the panel each pixel clock	R/W	0x0
8	LCDBE	0: Little endian operation is selected 1: Big endian operation is selected	R/W	0x0
7	LCD_TFT	0: Passive or STN display operation enabled 1: Active or TFT display operation enabled	R/W	0x0
6	LINE_INT_MASK	0: Mask the line interrupt going to the dedicated line lcd_line_int_n 1: Unmask the interrupt	R/W	0x0
5	LINE_INT_NIRQ_MASK	0: Interrupt is masked from going to nirq 1: Interrupt is unmasked	R/W	0x0
4	LOADMASK	0: Mask out the loaded palette interrupt going to nirq 1: Mask not active	R/W	0x0
3	LCDDONEMASK	0: Mask out the frame done interrupt going to nirq 1: Mask not active	R/W	0x0
2	VSYNC_MASK	0: Mask the VSYNC interrupt going to nirq 1: Unmask the interrupt	R/W	0x0
1	LCDBW	LCD monochrome 0: Color operation enabled 1: Monochrome operation enabled	R/W	0x0
0	LCD_ENABLE	LCD controller enable 0: LCD controller disabled 1: LCD controller enabled	R/W	0x0

Table 2−145. LCD Timing 0 Register (LCD_TIMING_0)

Base Address = 0xFFFE:C000, Offset = 0x04				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:24	HBP	Horizontal back porch. Program to value minus 1.	R/W	0x0
23:16	HFP	Horizontal front Porch. Program to value minus 1.	R/W	0x0
15:10	HSW	Horizontal sync pulse width. Encoded value used to specify number of pixel clock periods to pulse the line clock at the end of each line. Program to value minus 1.	R/W	0x0
9:0	PPL	Pixels per line. Encoded value used to specify number of pixels contained within each line on LCD display.Minimum is 16 pixel per line. Program to value minus 1.	R/W	0x0

Table 2−146. LCD Timing 1 Register (LCD_TIMING_1)

Base Address = 0xFFFE:C000, Offset = 0x08				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:24	VBP	Vertical back porch	R/W	0x0
23:16	VFP	Vertical front porch used to specify the number of line clock periods to add to the end of each frame.	R/W	0x0
15:10	VSW	Vertical sync pulse width. In active mode encoded value used to specify number of line clock periods to pulse the LCD_FP_O pin at the end of each frame after the end of frame wait period elapses. Program to value minus 1.	R/W	0x0
9:0	LPP	Lines per panel. Encoded value used to specify number of lines per panel. Program to value minus 1.	R/W	0x0

Table 2−147. LCD Timing 2 Register (LCD_TIMING_2)

Base Address = 0xFFFE:C000, Offset = 0x0C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:26	RESERVED	Reserved	R/W	0x0
25	PHSVS	HSYNC/VSYNC pixel clock control On/Off (Should be on only when in TFT mode) 0: HSYNC and VSYNC are driven on opposite edges of pixel clock than the pixel data 1: HSYNC and VSYNC are driven according to bit 24	R/W	0x0
24	PHSVSRF	Program HSYNC/VSYNC RISE OR FALL (to use, bit 25 must set to 1) 0: lcd_lp_o and lcd_fp_o are driven on falling edge of pixel clock. 1: lcd_lp_o and lcd_fp_o are driven on rising edge of pixel clock.	R/W	0x0
23	IEO	Invert output enable 0: lcd_ac_o pin is active high in active display mode 1: lcd_ac_o pin is active low in active display mode	R/W	0x0

Table 2−147. LCD Timing 2 Register (LCD_TIMING_2) (Continued)

Base Address = 0xFFFE:C000, Offset = 0x0C				
Bit	**Name**	**Function**	**R/W**	**Reset**
22	IPC	Invert pixel clock 0: Data is driven on LCD data lines on rising edge of lcd_cp_o 1: Data is driven on LCD data lines on falling edge of lcd_cp_o	R/W	0x0
21	IHS	Invert HSYNC 0: lcd_lp_o pin is active high and inactive low 1: lcd_lp_o pin is active low and inactive high	R/W	0x0
20	IVS	Invert VSYNC 0: lcd_fp_o pin is active high and inactive low 1: lcd_fp_o pin is active low and inactive high	R/W	0x0
19:16	ACBI	ac bias pin transitions per interrupt	R/W	0x0
15:8	ACB	ac bias pin frequency. Value used to specify the number of line clocks to count before transitioning the ac Bias pin	R/W	0x0
7:0	PCD	Pixel clock divisor. Value used to specify the frequency of the pixel clock based on the CPU clock frequency.	R/W	0x0

Table 2−148. LCD Status Register (LCD_STATUS)

Base Address = 0xFFFE:C000, Offset = 0x10				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:7	RESERVED	Reserved	R/W	0x0
6	LP	Loaded palette (read/clear only) 0: The palette is not loaded 1: The palette is loaded	R/C	0x0
5	FUF	FIFO underflow status 0: FIFO has not underrun 1: LCD dither logic not supplying data to FIFO at a sufficient rate.	R	0x0
4	LINE_INT	Line interrupt (read/clear only) 0: As long as display has not reached the programmed line yet. 1: Display reaches the user programmed line number	R/C	0x0
3	ABC	ac bias count status 0: ac bias transition counter has not decremented to zero 1: ac bias transition counter has decremented to zero	R/C	0x0
2	SYNC	Sync lost 0: Normal 1: Frame synchronization lost has occured	R	0x0

Table 2–148. LCD Status Register (LCD_STATUS) (Continued)

Base Address = 0xFFFE:C000, Offset = 0x10				
Bit	**Name**	**Function**	**R/W**	**Reset**
1	VS	VSYNC interrupt (read/clear only) 0: VSYNC interrupt has not generated 1: VSYNC interrupt has occured at the end of the frame	R/C	0x0
0	DONE	Frame done 0: LCD enabled 1: LCD disabled and the active frame has just completed	R	0x0

Table 2–149. LCD Sub-Panel Display Register (LCD_SUB_PANEL)

Base Address = 0xFFFE:C000, Offset = 0x14				
Bit	**Name**	**Function**	**R/W**	**Reset**
31	SPEN	Sub panel enable 0: Function disabled 1: Sub-panel function mode enabled	R/W	0x0
30	RESERVED	Reserved	R/W	0x0
29	HOLS	High or low signal. The field indicates the position of sub-panel compared to the LPPT value	R/W	0x0
28:26	RESERVED	Reserved	R/W	0x0
25:16	LPPT	Lines per panel threshold. This field defines the number of lines to be refreshed. LPPT is to be programmed to the value minus 1.	R/W	0x0
15:0	DPD	Default pixel data. DPD defines the default value of pixel data sent to panel for the lines until LPPT is reached or after passing the LPPT.	R/W	0x0

This register enables user to program the line number where he wants to genrate the interrupt

Table 2–150. Line Interrupt Register (LINE_INTERRUPT_REGISTER)

Base Address = 0xFFFE:C000, Offset = 0x18				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:10	RESERVED	Reserved	R	0x000000
9:0	LINE_INT_NUMBER	Line number at which line interrupt occurs	R/W	0x000

This register contains the line number currently being displayed by the panel

Table 2–151. Display Status register (DISPLAY_STATUS_REGISTER)

Base Address = 0xFFFE:C000, Offset = 0x1C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:11	RESERVED	Reserved	R	0x1FFFF
10:0	CURRENT_LINE_ NUMBER	Line number being displayed. As the number of lines can be programmed from 1 to 1024, the current line number varies between 0 and 1023	R	0x7FF

2.2.29 OCPI Registers

See ECN28 (Section 2.2.21) for changes.

Table 2–152 lists the OCPI registers. Table 2–153 through Table 2–160 describe the register bits.

Table 2–152. OCPI Registers

Base Address = 0xFFFE:C320				
Register	**Description**	**Bits**	**R/W**	**Offset**
ADDRFAULT	Address fault	32	R	0x00
MCMDFAULT	Master command fault	32	R	0x04
SINTERRUPT0	Interrupt configuration	32	R/W	0x08
ABORTTYPE	Type of abort	32	R	0x0C
SINTERRUPT1	Interrupt configuration	32	R/W	0x10
PROTECT	Enables protection for memory spaces	32	R/W	0x14
SECURE_MODE	Secure mode controls	32	R/W	0x18
DYNAMIC_POWER_DOWN	Dynamic Power Down	32	R/W	0x1C

Table 2–153. Address Fault Register (ADDRFAULT)

Base Address = 0xFFFE:C320, Offset = 0x00				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	ADDRESS	Address accessed by the master that causes an abort or error	R	0x0

Table 2–154. Master Command Fault Register (MCMDFAULT)

Base Address = 0xFFFE:C320, Offset = 0x04				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:3	RESERVED	Reserved. Must be 0	R	0x0
2:0	MCMD	Command that causes an abort or error	R	0x0

This generates interrupts at the OMAP boundary .

Table 2−155. Interrupt Configuration 0 Register (SINTERRUPT0)

Base Address = 0xFFFE:C320, Offset = 0x08				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:2	RESERVED	Reserved, must be 0	R/W	0x0
1	LEVELINTERRUPT	Level sensitive interrupt Sinterrupt_n signal has the same value with this bit. 0: Interrupt 1: No Interrupt	R/W	0x1
0	EDGEINTERRUPT	Edge-triggered interrupt Sinterrupt_n signal is asserted low for one cycle when this bit is set 0: Interrupt 1: No interrupt A read from this register always returns a 1, since it is auto cleared	R/W	0x1

Table 2−156. Type of Abort Register (ABORTTYPE)

Base Address = 0xFFFE:C320, Offset = 0x0C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:4	RESERVED	Reserved, must be 0	R	0x0
3	BURSTERROR	A burst access to the API or TIPB was requested	R	0x0
2	PROTECT	Address hit protected area	R	0x0
1	TRGABORT	Abort coming from accessed target	R	0x0
0	ADDDEC	Address decoding error(initiator sent unknown address)	R	0x0

The abort type register is cleared after it is read. In other words, reading the register once returns whatever bit corresponds to the abort type. On the next read, the register returns all 0. Also, a write to the register has no effect.

This generates interrupts at the OMAP boundary.

Table 2−157. Interrupt Configuration 1 Register (SINTERRUPT1)

Base Address = 0xFFFE:C320, Offset = 0x10				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:2	RESERVED	Reserved,must be 0	R/W	0x0
1	LEVELINTERRUPT	Level sensitive Interrupt Sinterrupt_n signal has the same value with this bit. 0: Interrupt 1: No Interrupt	R/W	0x1
0	EDGEINTERRUPT	Edge-triggered Interrupt Sinterrupt_n signal is asserted low for one cycle when this bit is set 0: Interrupt 1: No interrupt	R/W	0x1

Enables protection for memory spaces.

Table 2−158. Memory Space Protection Register (PROTECT)

Base Address = 0xFFFE:C320, Offset = 0x14				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:8	RESERVED	Reserved, must be 0	R/W	0x0
7	API	Access to API is prohibited from initiator.	R/W	0x0
6	TIPBPUB	Access to MPU TIPB public bus is prohibited from initiator.	R/W	0x0
5	RHEAPRIV	Access to MPU TIPB private bus is prohibited from initiator.	R/W	0x0
4	OCPMULT	Access to OCP multibank is prohibited from initiator.	R/W	0x0
3	OCPT2	Access to OCPT2 is prohibited from initiator.	R/W	0x0
2	OCPT1	Access to OCPT1 is prohibited from initiator.	R/W	0x0
1	EMIFF	Access to EMIFF is prohibited from initiator.	R/W	0x0
0	EMIFS	Access to EMIFS is prohibited from initiator.	R/W	0x0

Reset value is equal to the value of static_reset_protect_mode.

Table 2−159. Secure Mode Control Register (SECURE_MODE)

Base Address = 0xFFFE:C320, Offset = 0x18				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:7	RESERVED	Reserved. Must be 0	R/W	0x0
6	API	In secure mode: 1: Access to MPU is prohibited from initiator 0: Access is allowed	R/W	0x1

Table 2−159. Secure Mode Control Register (SECURE_MODE) (Continued)

Base Address = 0xFFFE:C320, Offset = 0x18				
Bit	**Name**	**Function**	**R/W**	**Reset**
5	RHEAPUB	In secure mode: 0: Access is allowed 1: Access to MPU TIPB public bus is prohibited from Initiator; MPU TIPB private bus cannot be accessed regardless of setting of this bit.	R/W	0x1
4	OCPTMULT	In secure mode: 0: Access is allowed 1: Access to OCP multi-bank is prohibited from Initiator	R/W	0x1
3	OCPT2	In Secure mode: 1:Access to OCPT2 is prohibited from Initiator	R	0x1
2	OCPT1	In secure mode: 0: Access is allowed 1: Access to OCPT1 is prohibited from Initiator	R/W	0x1
1	EMIFF	In secure mode: 0: Access is allowed 1: Access to EMIFF is prohibited from Initiator	R/W	0x1
0	EMIFS	In secure mode: 0: Access is allowed to CS1-CS3 1: Access to EMIFS CS1-CS3 is prohibited from Initiator; CS0 cannot be accessed regardless of setting of this bit.	R/W	0x1

When not in secure mode, these register values are ignored. When in secure mode, EMIFS cs0 and TIPB private access is not allowed, regardless of register values. Every other target secure mode is determined by the register value.

Enable/Disable power down mode on OCPI to save power.

Table 2−160. Dynamic Power Down Register (DYNAMIC_POWER_DOWN)

Base Address = 0xFFFE:C320, Offset = 0x1C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:1	RESERVED	Reserved	R/W	0x00000000
0	AUTO_GATED_CLK	0: Autoclock gating is disabled. 1: Autoclock gating is enabled to save power.	R/W	0x0

Chapter 3

Camera Interface

This chapter describes the camera interface implemented in the OMAP850 multimedia processor.

Topic **Page**

3.1 Camera Parallel Interface 3-2

3.1 Camera Parallel Interface

A 32-bit camera interface connects a camera module to the MPU peripheral bus of the OMAP850 device. The interface handles multiple image formats synchronized on vertical and horizontal synchronization signals. Data transfer between the camera and the interface can be done synchronously or asynchronously. The data is stored in a buffer to be sent over the peripheral bus, using the DMA mode or the CPU mode (bypass mode).

The interface supports 8-bit parallel image data ports and horizontal/vertical signal ports separately (stand-alone synchronous method). The camera interface has a DMA port.

3.1.1 Functional Architecture

The camera architecture consists of four functional blocks:

- Buffer: Stores the data word received from the camera module and transfers it to the MPU peripheral bridge, using the DMA mode or the CPU mode. It contains a 128-word FIFO.

  The 8-bit data received from the camera module is packed into 32 bits. A 128-bit-deep FIFO is implemented to provide local buffering of the data and to control the DMA request when the camera interface is enabled in DMA mode. The main goals of this mode are:

  - To discharge the CPU of the data transfer
  - To reduce the real time constraints of the DMA read (FIFO buffering part)
  - To group the x DMA accesses in only one time slot (FIFO block part)

  The user can, however, forward a direct transfer to the CPU in bypass mode by disabling the DMA request line.

- Clock divider: Manages the clock division and handles the external clock generation for synchronous/asynchronous mode gating

- Interrupt generator: Generates an interrupt to indicate the start and end of frame, start and end of image, and FIFO overflow

- Registers: Connect status, control, and data 32-bit registers

3.1.1.1 Camera I/F Clocks

The camera I/F functional clock is either the ARMPER_CK divided down (see the CONTROLCLK.FOCSMOD bits for divide down options) or CAM.LCLK (input pin D17).

The camera I/F interface clock is ARMPER_CK.

3.1.1.2 Camera Data Validation

The incoming byte on CAM_D can be latched on the rising or falling edge of CAM.LCLK generated by the camera itself. The POLCLK bit in the clock control register selects the polarity of CAM.LCLK.

The camera interface must be programmed so that data is always captured opposite the launch edge. For example, if data is latched by the sensor on the rising edge of CAM.LCLK, the interface must be configured to catch the data on the falling edge of CAM.LCLK.

The high level of the vertical synchronous and horizontal synchronous signals indicates that the data is valid on CAM_D. This level is registered in VSTATUS and HSTATUS, which are updated at edge detection of vertical and horizontal synchronous signals.

Figure 3−1 shows the image data transfer, and Figure 3−2 shows the timing chart.

Figure 3−1. Image Data Transfer

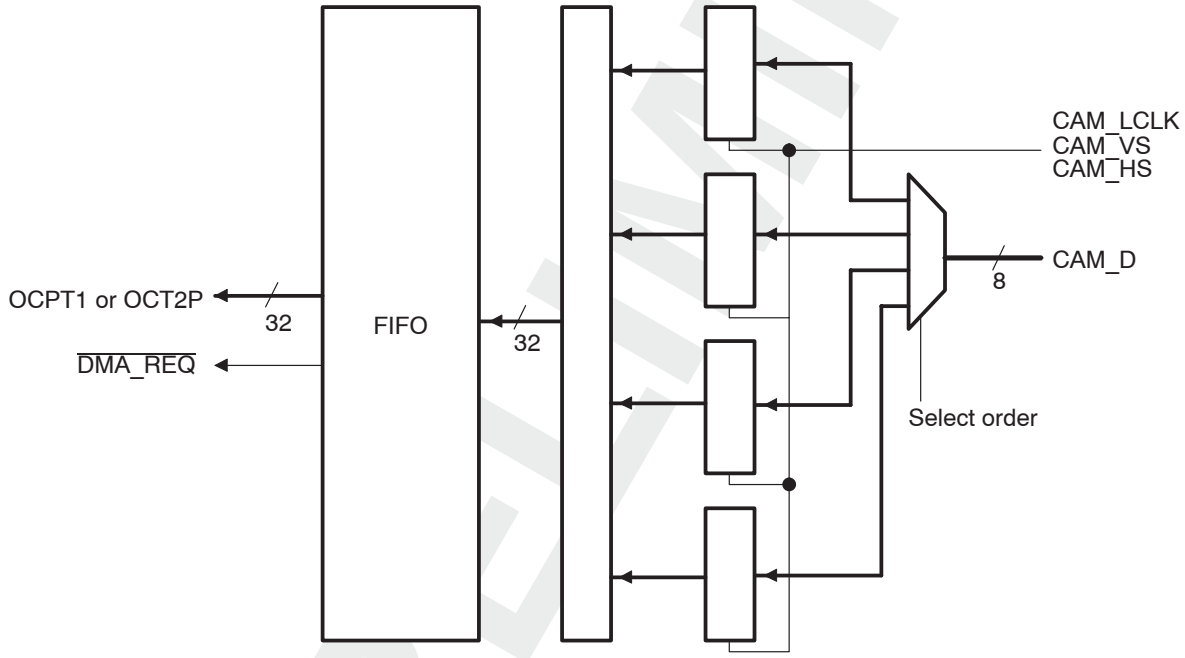
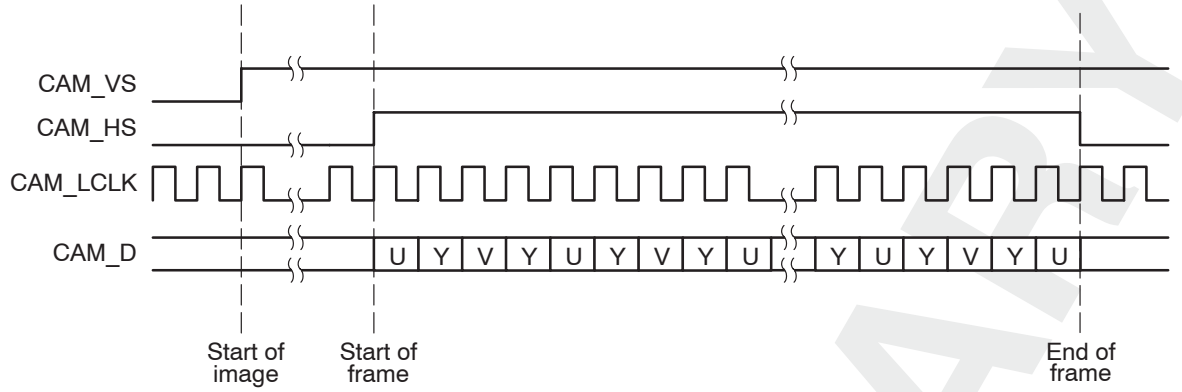


Figure 3–2. Timing Chart of Image Data Transfer (POLCLK = 1)



The clock can be gated during the VSYNC and/or HSYNC blanking periods, but it must be let to run because a process based on LCLK clears all internal resynchronization registers while VSYNC or HSYNC is low, before starting a new frame or new image. This mechanism prevents the FIFO from retaining any remaining (and likely garbage) data left over from a previous frame, which could corrupt the data of a new frame.

If either CAM_VS or CAM_HS goes inactive before receiving all four bytes, the data in buffers is cleared by the active CAM.LCLK edge and is not written into FIFO.

3.1.1.3 Autostart

Autostart is a protection function that prevents a start of capture during an image transfer. Autostart is launched after enabling the LCLK and waits for the next inactive level of CAM_VS to enable the data capture so that the transfer starts at the beginning of the image.

Note:

If a reset FIFO occurs (see Section 3.1.1.4) while the interface is latching data, the capture is automatically disabled and the autostart function is enabled.

3.1.1.4 Reset FIFO

An active-high reset FIFO is implemented at the RAZ_FIFO bit (18) of the camera mode register. This feature clears any remaining data in the FIFO before starting a new transfer. It also resets all status and control signals around the FIFO, such as the read and write pointers, the FIFO full interrupt, the FIFO peak counter, and the 32-bit resynchronization registers.

3.1.1.5 Set of Order

Each four bytes received from the camera must be packed and can be swapped to follow the order YUV specified in the camera mode register by ORDERCAMD. Figure 3–3 and Figure 3–4 show the camera data order not swapped and swapped, respectively.

Figure 3–3. Order of Camera Data on OCPT1 (Not Swapped)

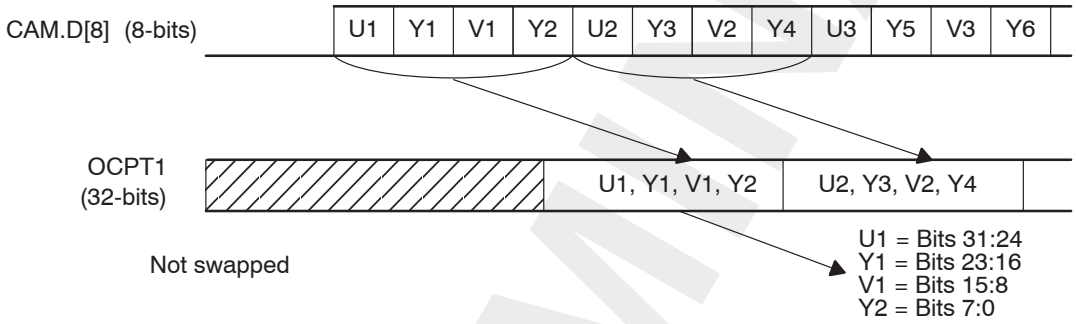
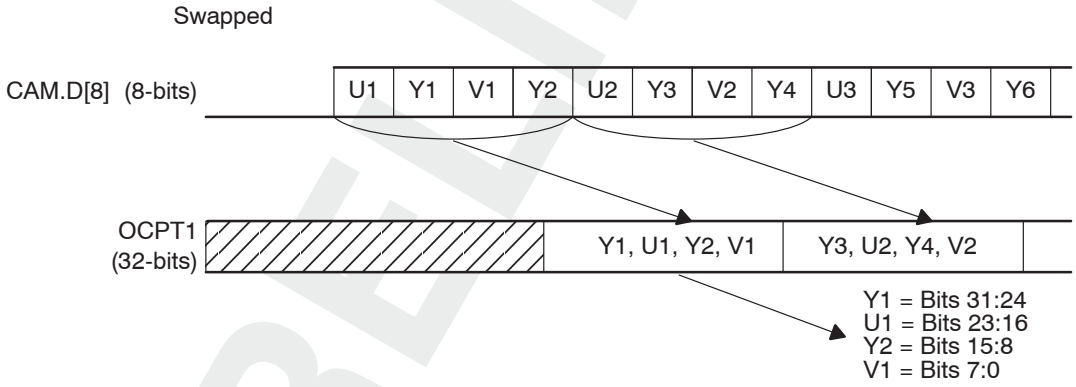


Figure 3–4. Order of Camera Data on OCPT1 (Swapped)



3.1.1.6 FIFO Buffer (128 x 32)

A write access is applied to the FIFO for each 32-bit word received. When the write FIFO counter reaches the trigger level, an interrupt request can be generated. The trigger level is programmable.

In DMA mode, the threshold can be programmed between 1 and 128, but the DMA must be set up to read the threshold amount out of FIFO per the DMA request issued by the camera interface. Otherwise, the locking mechanism is never rearmed, and it prevents DMA requests from being issued after every read.

A pulse on the DMA request (see Figure 3–5 and Figure 3–6) occurs when the number of words in the FIFO is above the threshold. The DMA request occurs if the number of remaining words is above the threshold and the system DMA has completed the transfer (number of words read by the DMA = threshold).

The camera FIFO continues to fill (up to its maximum 128 values) when an interrupt or DMA request has been generated but not yet responded to. When a data value is read from the camera FIFO, another IRQ or DMA request is immediately generated, as long as the amount of data present in the FIFO is above the trigger level.

Figure 3–5. DMA Request

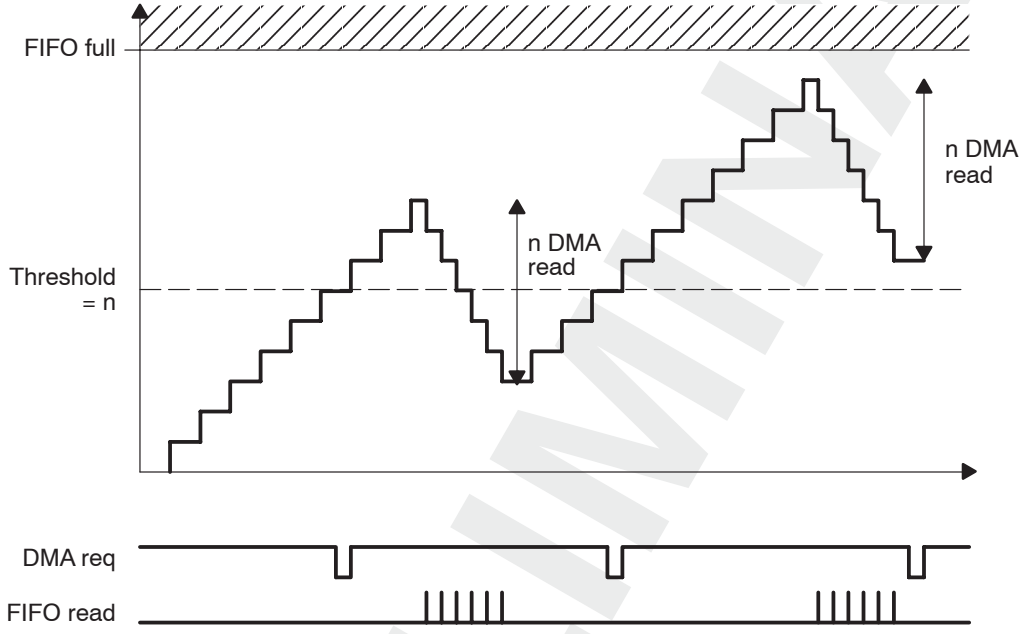
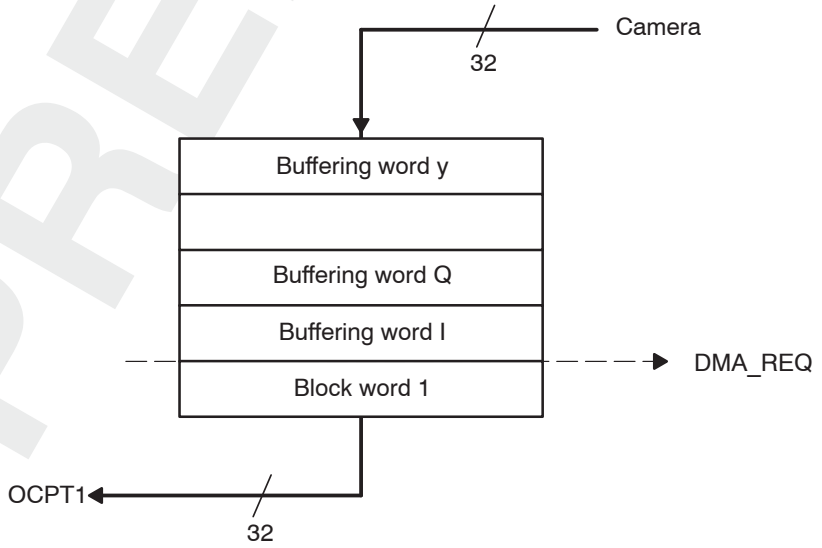


Figure 3–6. FIFO Buffer Parts



3.1.1.7 Clock Divider

The clock divider takes the OCP T2 clock TC2_CK as clock source to generate the external clock CAM.EXCLK. The division factor is programmable in the clock control register through FOSCMOD (see Table 3–3).

It is assumed that the switch is made when CAM.EXCLK is disabled (glitch protection).

The clock divider also allows disabling the external clock by setting the CAMEXCLK_EN bit.

3.1.1.8 Interrupt Generator

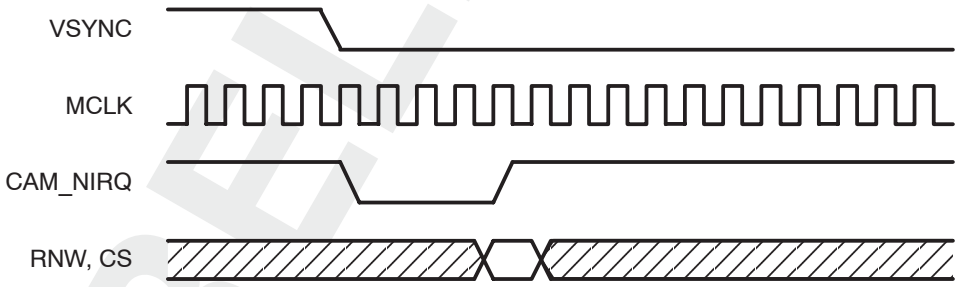
The interrupt generator handles six cases of interrupt:

- Data transfer interrupt. One IRQ is generated per word received.
- HSYNC rising edge (start of frame)
- HSYNC falling edge (end of frame). See Figure 3–7.
- VSYNC rising edge (start of image)
- VSYNC falling edge (end of image)
- FIFO overflow

Each case is registered by activating (high) one of the six interrupt register bits to indicate the origin of the interrupt. However, the interrupt mask register can disable the source of the interruption.

Only one line of interrupt is used to ask for a read of the interrupt register. When the read occurs, the register is automatically reset and the interrupt signal is released.

Figure 3–7. IRQ Generated on VSYNC Falling Edge



3.1.1.9 DMA Procedure

A typical procedure to perform the data transfer by DMA is as follows:

1) Rising edge of VSYNC sends an interrupt to ARM926EJS to alert the system DMA that a start of image has occurred. The system DMA is programmed to move one complete image of data, and then give an interrupt when complete.

2) High level of HSYNC and proper clock edge start the first data transfer from the camera to the OMAP850 camera interface. After the first two pixels of data are received (8 bits x 4 transfers = 32 bits), a DMA request is made. The system DMA moves the 32-bit data to a predefined SDRAM location.

3) The camera, the OMAP850 device camera interface, and the system DMA continue the transfer of data. That is, 352/2 * 288 = 50688 transfers for a camera interface image format. After the full image is transferred, the DMA sends an interrupt to the ARM926EJS to signal that the end of frame occurred.

The camera interface and system DMA can be configured in many ways to move the data, but in this sequence the interrupt load on the ARM926EJS is minimal.

3.1.1.10 Camera IF Registers

The camera interface contains seven registers for communication between the OCPT1 and camera module. These registers mainly control clock generation, interrupt request, and status register (see Section 3.1.1.11).

Table 3−1 shows the default configuration at reset.

Table 3−1. Default Configuration at Reset

Item	Function
ORDERCAMD	Not swapped
MASK	Interrupts on VSYNC and HSYNC disabled
FOSCMOD	Division rate for CAM.EXCLK = 1
POLCLK	Data latched on rising edge of CAM.LCLK
CAMEXCLK_EN	CAM.EXCLK disabled
MCLK_EN	Internal clock disabled
APLL_EN	APLL clock source disabled
THRESHOLD	Trigger level = 1 word

3.1.1.11 Camera Interface Registers

Table 3–2 lists the camera interface registers. Table 3–3 through Table 3–9 describe the bits of the individual registers.

Table 3–2. Camera Interface Registers

Base Address = 0x2004 0000 (OCP T1), 0x4000 0000 (OCP T2)				
Register	**Description**	**R/W**	**Size**	**Offset**
CTRLCLOCK	Clock control	R/W	32 bits	0x00
IT_STATUS	Interrupt source status	R	32 bits	0x04
MODE	Camera interface mode configuration	R/W	32 bits	0x08
STATUS	Status	R	32 bits	0x0C
CAMDATA	Image data	R	32 bits	0x10
GPIO	Camera interface GPIO (general-purpose input/ output)	R/W	32 bits	0x14
PEAK_COUNTER	FIFO peak counter	R/W	32 bits	0x18

Table 3–3. Clock Control Register (CTRLCLOCK)

Base Address = 0x2004 0000 (OCP T1), 0x4000 0000 (OCP T2)				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:8	RESERVED	This field is reserved (unknown value after reset).	R/W	ND
7	LCLK_EN	0: Disables 1: Enables incoming CAM.LCLK	R/W	0x0
6	RESERVED	Reserved	R/W	0x0
5	MCLK_EN	0: Disables 1: Enables internal clock of interface	R/W	0x0
4	CAMEXCLK_EN	0: Disables 1: Enables CAM.EXCLK	R/W	0x0
3	POLCLK	Sets polarity of CAM.LCLK: 0: Data latched on rising edge 1: Data latched on falling edge	R/W	0x0
2:0	FOSCMOD	Sets the frequency of the CAM.EXCLK clock 000: TC2_CK / 8 001: TC2_CK / 3 010: TC2_CK / 16 011: TC2_CK / 2 100: TC2_CK / 10 101: TC2_CK / 4 110: TC2_CK / 12 111: Inactive	R/W	0x00

The MCLK_EN bit gates the master clock of the camera interface to disable the clock when switching between two clock domains or to save power consumption when the camera module is not used. To clear

PEAK_COUNTER, read all data in FIFO and then write PEAK_COUNTER with 0.

Table 3–4. Interrupt Source Status Register (IT_STATUS)

Base Address = 0x2004 0000 (OCP T1), 0x4000 0000 (OCP T2)				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:6	RESERVED	Reserved bits	R	0xX
5	DATA_TRANSFER	Data transfer status. Set to 1 when trigger is reached. Reset by reading IT_STATUS if no event in the meantime.	R	0x0
4	FIFO_FULL	Detect rising edge on FIFO full flag. Reset by reading IT_STATUS if no event in the meantime.	R	0x0
3	H_DOWN	Flag for horizontal synchronous falling edge occurred. Reset by reading IT_STATUS if no event in the meantime.	R	0x0
2	H_UP	Flag for horizontal synchronous rising edge occurred. Reset by reading IT_STATUS if no event in the meantime.	R	0x0
1	V_DOWN	Flag for vertical synchronous falling edge occurred. Reset by reading IT_STATUS if no event in the meantime.	R	0x0
0	V_UP	Flag for vertical synchronous rising edge occurred. Reset by reading IT_STATUS if no event in the meantime.	R	0x0

Table 3–5. Camera Interface Mode Configuration Register (MODE)

Base Address = 0x2004 0000 (OCP T1), 0x4000 0000 (OCP T2)				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:19	RESERVED	Reserved bits	R/W	0xX
18	RAZ_FIFO	When 1: Clears data in the FIFO; reinitializes read and write pointers; clears FIFO full interrupt, FIFO peak counter; and resynchronizes.	R/W	0x0
17	EN_FIFO_FULL	0: Disables 1: Enables interrupt on FIFO_FULL	R/W	0x0
16	EN_NIRQ	0: Disables 1: Enables data transfer interrupt (bypass DMA mode)	R/W	0x0
15:9	THRESHOLD	Programmable DMA request trigger value. DMA request is made when FIFO counter is equal to the threshold value. Currently, set this field to 1 in DMA mode.	R/W	0x0000001
8	DMA	Enables DMA mode when 1	R/W	0x0
7	EN_H_DOWN	Enables interrupt on HSYNC falling edge. Active when 1.	R/W	0x0

Table 3–5. Camera Interface Mode Configuration Register (MODE) (Continued)

Base Address = 0x2004 0000 (OCP T1), 0x4000 0000 (OCP T2)				
Bit	**Name**	**Function**	**R/W**	**Reset**
6	EN_H_UP	Enables interrupt on HSYNC rising edge. Active when 1.	R/W	0x0
5	EN_V_DOWN	Enables interrupt on VSYNC falling edge. Active when 1.	R/W	0x0
4	EN_V_UP	Enables interrupt on VSYNC rising edge. Active when 1.	R/W	0x0
3	ORDERCAMD	Sets order of 2 consecutive bytes received from camera (YUV format). Not swapped when 0, swapped when 1.	R/W	0x0
2:1	IMGSIZE	Sets image size:. - CIF when 00 - QCIF when 01 - VGA when 10 - QVGA when 11 Currently, these bits have no effect on the operation of the camera interface.	R/W	0x00
0	CAMOSC	0: Set synchronous mode 1: Set asynchronous mode Currently, this has no effect on the camera interface.	R/W	0x0

Table 3–6. Status Register (STATUS)

Base Address = 0x2004 0000 (OCP T1), 0x4000 0000 (OCP T2)				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:2	RESERVED	Reserved bits	R	0xX
1	HSTATUS	CAM_HS status (edge detection)	R	0x0
0	VSTATUS	CAM_VS status (edge detection)	R	0x0

Table 3–7. Camera Interface GPIO Register (GPIO)

Base Address = 0x2004 0000 (OCP T1), 0x4000 0000 (OCP T2)				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:1	RESERVED	Reserved bits	R/W	0xX
0	CAM_RST	Reset for camera module	R/W	0x0

Table 3–8. Image Data Register (CAMDATA)

Base Address = 0x2004 0000 (OCP T1), 0x4000 0000 (OCP T2)				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	CAMDATA	Image data from FIFO	R	0x0

Table 3–9. FIFO Peak Counter Register (PEAK_COUNTER)

Base Address = 0x2004 0000 (OCP T1), 0x4000 0000 (OCP T2)				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:7	RESERVED	Reserved	R/W	Unknown
6:0	PEAK_COUNTER	Maximum number of words written to FIFO during the transfer since the last clear to zero	R/W	0x0000000

3.1.2 Clock Switching Procedures

3.1.2.1 CAM.EXCLK Switch Protocol

The CAM.EXCLK switch protocol is required for any change of the CAM.EXCLK frequency value. First, disable the MCLK clock source and the APLL clock source in clock registers, as follows:

1) Disable CAM.EXCLK (CAMEXCLKEN register bit = 0).

2) Change CAM.EXCLK value (FOSCMOD = new FOSCMOD).

3) Enable CAM.EXCLK (CAMEXCLKEN register bit = 1).

The selection of clock output for the camera sensor is done with the CAM_EXCLK_SELECT bit (see Table 5–17, *Spare 1 Register for ECO (SPARE1)*).

3.1.2.2 CAM.LCLK Switch Protocol

Bit 3 of the clock control register (POLCLK) sets the polarity of CAM.LCLK. CAM.LCLK must be disabled before selecting the rising or falling edge.

1) Disable CAM.LCLK (LCLK_EN = 0).

2) Set the new polarity (POLCLK = 1 or 0).

3) Enable CAM.LCLK (LCLK_EN = 1).

Chapter 4

LCD Interface

This chapter discusses the HR-TFT LCD interface module of the OMAP850 multimedia processor.

Topic **Page**

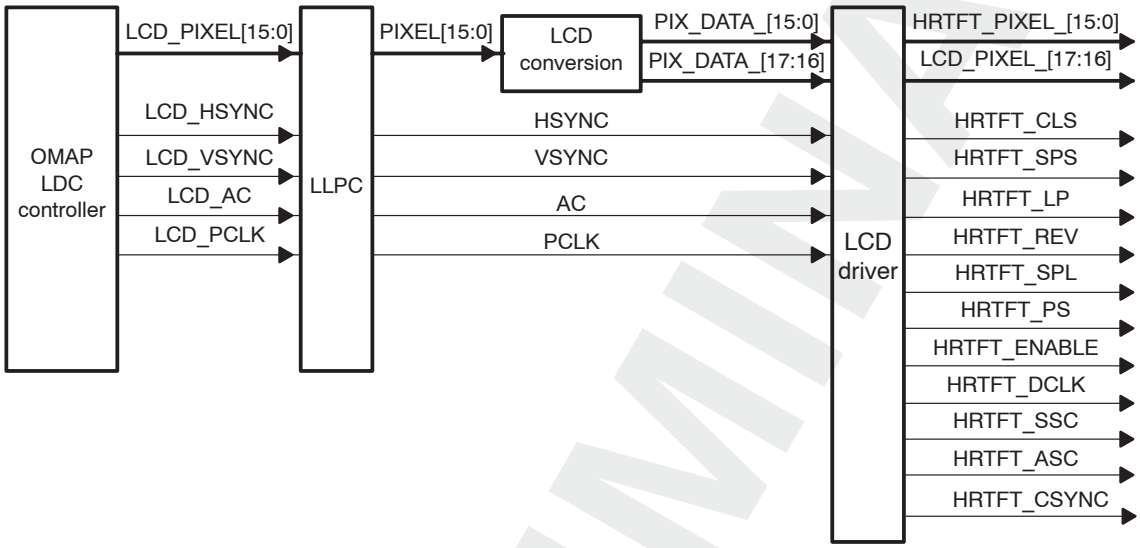
4.1	Introduction	4-2
4.2	Features and Supported Panel Types	4-3
4.3	Functional Description	4-5
4.4	Registers	4-10
4.5	Programming Model	4-17

4.1 Introduction

The LCD driver sub-block contains the logic required to post-process the signals coming out of the OMAP internal LCD controller. This logic generates signals required to interface the OMAP LCD controller with HR-TFT-like LCD panels (see Figure 4–1).

For details on the working and register set of the LCD controller, see Chapter 8, *LCD Controller*, of the OMAP730 TRM.

Figure 4–1. OMAP850 HR-TFT Interface High-Level Diagram



4.2 Features and Supported Panel Types

- ❑ HR-TFT display panels support
- ❑ Support for 16bpp/18bpp
- ❑ Bypass functionality for HR-TFT mode
- ❑ Data pixel blanking feature
- ❑ Programmable timing control signals to support different panel requirements
- ❑ Power management features

Table 4–1. Supported Panel Types

LQ035Q7DB02	All, TFT Active Matrix	3.5 inch	QVGA (240 X 320)	Color (6-bit), data input (digital), integral touch panel, led backlight, low power, temperature ranges (operating temperature -10c to +60c), display orientation (portrait), sunlight readable, panel mode (transflective)
LQ035Q7DB03	TFT Active Matrix	3.5 inch	QVGA (240 X 320)	Power (365 mW), color (6-bit), panel mode (advanced tft), data input (digital), LED backlight, temperature ranges (operating temperature -10c to +60c), display orientation (portrait), sunlight readable, panel mode (transflective)
LQ035Q7DH01	TFT Active Matrix	3.5 inch	QVGA (240 X 320)	Color (6-bit), data input (digital), integral touch panel, LED backlight, low power, display orientation (portrait), power, sunlight readable
LQ038Q7DB03	TFT Active Matrix	3.8 inch	QVGA (240 X 320)	Color (6-bit), panel mode (advanced TFT), data input (digital), LED backlight, low power, display orientation (portrait), sunlight readable, ultra-low power < 1.0 W
LQ038Q7DB01	TFT Active Matrix	3.75 inch	320 X 480	262,144 colors , data input (digital), integral touch panel, LED backlight, low power, panel mode (transflective)
LQ030	TFT Active Matrix	2.99 inch	320 X 320	262,144 colors , data input (digital), integral touch panel, LED backlight, low power, panel mode (transflective)
LS037V7DD01	TFT Active Matrix	3.7 inch	VGA/ QVGA	Data input (digital), integral touch panel, LED backlight, low power, temperature ranges (operating temperature -20c to +70c), sunlight readable
LS040V7DD01	TFT Active Matrix	4.0 inch	VGA/ QVGA	Data input (digital), integral touch panel, LED backlight, low power, temperature ranges (operating temperature -20c to +70c), sunlight readable, panel mode (transflective)

4.2.1 Module Interface

Table 4–2 lists the HR-TFT LCD interface signals.

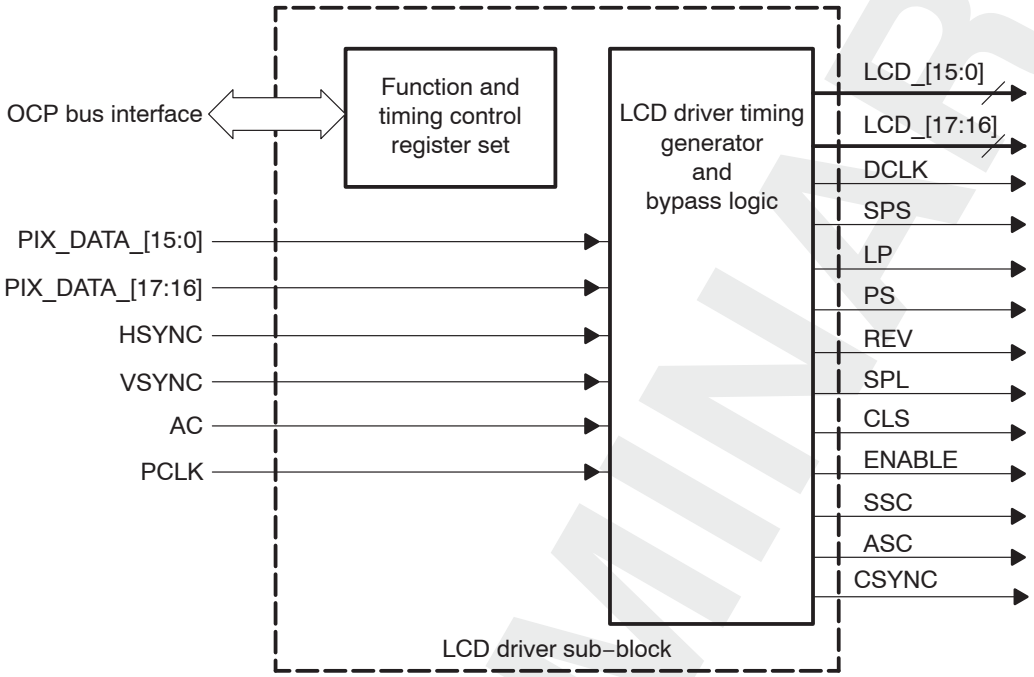
Table 4–2. HR-TFT LCD Interface Signals

Pin Name	Pin Direction	Description
HRTFT_CLS	OUT	Clock and sync signal is for row gate drivers. The repetition rate is similar to the horizontal sync signal, but it carries different setup and hold requirements
HRTFT_SPS	OUT	When HR-TFT sub-block is bypassed, Vsync signal from OMAP LCD controller is directly output on this signal. When HR-TFT is enabled, it functions as SPS. Start panel sync is similar to vertical sync signal which is pulse activated when one page or frame of data has been transmitted to LCD
HRTFT_LP	OUT	When in bypass mode, the HSYNC from OMAP LCD controller is directly output on this signal. When in HR-TFT mode, line pulse, similar to horizontal synchronization signal, is transmitted on this signal.
HRTFT_PS	OUT	Is the source driver control signal. The repetition rate and waveshape Is the same as CLS, but with reversed polarity
HRTFT_SPL	OUT	Start pulse left/right(SPL/R) is source driver start signal. Its repetition is tied to horizontal sync signal, and it occurs just after the HSYNC pulse, to indicate first valid data for line
HRTFT_REV	OUT	REV signal toggles every time horizontal synchronization signal is sent to panel and is on frame boundary
HRTFT_CSYNC	OUT	This output of an *xor* gate (2 inputs) has HSYNC and VSYNC inputs and is used only for analog LCD panels.
HRTFT_ENABLE	OUT	This signal enables signal HR-TFT modes and indicates to panel when valid display data is available. When bypass mode is enabled, ac_bias signal from OMAP LCD controller is directly output on this signal.
HRTFT_DCLK	OUT	In bypass mode, clock signal from OMAP LCD controller is directly output on this signal. In HR-TFT mode, dot clock is output on this signal and is clock source for LCD panel.
HRTFT_ASC	OUT	Analog switch control signal
HRTFT_SSC	OUT	Source signal control
HRTFT_PIXEL_[15:0]	OUT	Pixel data bus
LCD_PIXEL_[17:16]	OUT	Pixel data bus

4.3 Functional Description

Figure 4–2 shows the HR-TFT interface functionality.

Figure 4–2. HR-TFT LCD Interface Functional Diagram



The LCD driver sub-block generates HR-TFT-compatible timing signals to interface with a HR-TFT panel. The sub-block comprises a register set and a timing generator.

- The register set consists of a functional control register and timing control registers, one for each of the timing control signals as required by a LCD panel.
- The timing generator generates the timing control signals taking inputs from the register set and from the OMAP LCD controller.

The LCD driver block works basically in bypass mode and non-bypass mode.

4.3.1 Bypass Mode

When the LCD_DRIVER sub-block is configured in bypass mode (specified by bit 0 of the functional control register0, LCDDRV_FnCtl0(0)), the LCD_DRIVER timing signals are rendered inactive and the pixel_data along with HSYNC, VSYNC, PIXEL_CLK, AC_BIAS from the OMAP LCD controller is directly output on the device pins.

Table 4–3 lists the mapping of the OMAP LCD pins to the LCD driver output pins in bypass mode. In bypass mode the signals in column 1 are directly assigned to signals in column 2.

Table 4–3. LCD Driver Sub-Block Mapping (Bypass Mode)

OMAP LCD Controller Outputs	Mapping to LCD Driver Sub-Block Outputs in Bypass Mode
PIX_DATA_[15:0]	HRTFT_PIXEL_[15:0]
PIX_DATA_[17:16]	LCD_PIXEL_[17:16]
HSYNC	HRTFT_LP
VSYNC	HRTFT_SPS
AC	HRTFT_ENABLE
PCLK	HRTFT_DCLK

4.3.1.1 Non-Bypass Mode (HR-TFT Mode)

When the bypass functionality is disabled, the LCD driver timing signals are active to drive an HR-TFT LCD panel (specified by bit 0 of the functional control register0, LCDDRV_FnCtl(0))

The control signals are as follows:

- The DCLK (dot clock) is the clock source for the LCD. In HR-TFT panels, it is a square wave active only when data is valid. It is not a continuous waveform as in STN or TFT panels. But like STN and TFT panels, the clock frequency must be correct and conform to the panel specification to avoid flickering.
- The line pulse (LP) is similar to a horizontal synchronization signal, which is a pulse activated when one line of data has been transmitted to the LCD
- The SPS (StartPanelSync) signal is similar to a vertical synchronization signal, which is a pulse, activated when one page or frame of data has been transmitted to the LCD.
- The CLS (Clock and Sync) is the clock signal for row gate drivers. The repetition rate is similar to the horizontal synchronization signal, but is carries different setup and hold times that depend on the LCD panel that it interfaces to.
- The PS (power saving) is the source driver control signal that is used for power saving.

  When PS is low, the source driver output is enabled and data is applied to the LCD display.

  When PS is high, the source driver output is Z. The repetition rate is similar to the CLS signal, but with reverse polarity.

  The timing on this signal may be different depending on whether the LCD module is in display period or in non-display period.

  - In display period, the timing signal PS1 or PS2 (controlled by bit 2 of functional control register0, LCDDRV_FnCtl0(2)) should be output.
  - In non-display period, PS3 (controlled by bit 3 of functional control register0, LCDDRV_FnCtl0(3)) should be output.

So for display/non-display periods, combinations PS1/PS3 or PS2/PS3 could be generated.

- The SPL/SPR (start pulse left and start pulse right) is a trigger signal for the driver IC shift registers on the LCD panel glass. The shift register has dual scan directions: left and right.

  The LCD panel module has 2 trigger signals, SPL for scanning to left and SPR for scanning to right:

  - When using SPL, SPR line must be open or high impedance.

    SPL is the source driver start signal. Its repetition rate is tied to the horizontal synchronization signal, and it occurs just after the horizontal synchronization pulse to indicate the first valid data for the line.

    When SPL is used for starting the signal output for source driver, the scan begins on the left side of the panel.

  - When usinbg SPR, SPL must be open or high Impedance.

    SPR has the same function and repetition rate as SPL. These two signals are mutually exclusive, in that they are used in conjunction with LBR (horizontal scan direction signal that is an input on the LCD panel) to decide the horizontal scan direction.

    When SPR is used, the scan begins on the right side of the panel.

  Table 4–4 summaries this functionality (with respect to the LCD panel).

Table 4–4. HR-TFT Horizontal Scanning Direction Selection

LBR	SPL	SPR	SCAN Direction
1	Input	Output	Normal (left to right)
0	Output	Input	Reversed (right to left)

- The CSYNC composite signal is generated from an *xor* function that has HSYNC and VSYNC inputs from the OMAP LCD controller and is used for analog panels.

  Note: the CSYNC signal is used for analog panels while the SPL_SPR signal is used for digital panels. Both these signals are output on the same device pin, and proper selection must be made at the device I/O muxing block.

- The REV signal provides reversing of the voltage across the liquid crystal cell, on a line-by-line basis. This signal toggles every time a horizontal synchronization signal is send to the panel. It changes phase of every line pulse and on every frame pulse.

- The ENABLE signal indicates to the LCD panel when the valid display data is available. Active pixel data is driven on the LCD panel during the active edge of this signal.

- The analog switch control signal (ASC) is a special requirement for the sharp LCD panel.

- The source signal control signal (SSC) is also a special requirement for sharp LCD panels.

Table 4–5 maps the name of these control signals with their corresponding HR-TFT sub-block outputs in non-bypass or HR-TFT mode.

Table 4–5. LCD Driver Sub-Block Output Mapping of Control Signals (HR-TFT Mode)

HR-TFT Control Signals	Mapping to HR-TFT Outputs in Non-Bypass Mode or HR-TFT Mode
DCLK	HRTFT_DCLK
LP	HRTFT_LP
SPS	HRTFT_SPS
CLS	HRTFT_CLS
PS	HRTFT_PS
SPL/SPR	HRTFT_SPL
CSYNC	HRTFT_CSYNC
REV	HRTFT_REV
ENABLE	HRTFT_ENABLE
SSC	HRTFT_SSC
ASC	HRTFT_ASC

Note: In HR-TFT mode the control signals in column 1 are available on the signal names of column 2.

The data signals are as follows:

- LCD_[15:0]
- LCD_[17:16]

There are a total of 18 data signals corresponding to LCD_[15:0] and LCD_[17:16]. The LCD conversion module provides a 16-bit to 18-bit data conversion to the OMAP LCD interface. It supports two operating modes:

- 16-bit LCD
- 18-bit LCD

Setting a dedicated bit in its control register switches the mode. The current status of operation is shown via a status register:

- When the 16-bit LCD mode is used, the module operates in bypass mode, and all the 16-bit LCD data pixels coming from OMAP are directly provided to the external LCD interface. These signals are available on LCD_[15:0].
- When the 18-bit LCD mode is used, the 16-bit LCD pixel signal is converted to an 18-bit LCD pixel signal. The 18-bit LCD pixel format adds the two MSBs equal to the two green-color LSBs. These two signals are

available on LCD_[17:16], and in addition to LCD_[15:0], make a total of 18-bit LCD data pixels.

The pixel blanking feature is as follows:

When in TFT or HR-TFT mode the output pixel data bus *(LCD_[15:0] & LCD_[17:16])* can set to zero during the non-display periods on the screen. This feature is controlled by bit 1 of the functional control register. The non-display period corresponds to the time intervals when active video pixel data is not being transmitted.

This feature is active in HR-TFT and TFT modes.

4.4 Registers

Table 4–6 lists the LCD driver control and timing registers. Table 4–7 through Table 4–20 describe the register bits.

Table 4–6. LCD Driver Control and Timing Registers

Base Address = 0xFFFC:2000			
Name	**Description**	**R/W**	**Offset**
LCDDRV_FnCtl0	LCDDRV functional control 0	R/W	0x00
LCDDRV_FnCtl1	LCDDRV functional control 1	R/W	0x04
LCDDRV_TimLp	LCDDRV timing_Lp control	R/W	0x08
LCDDRV_TimCls	LCDDRV timing_Cls control	R/W	0x0C
LCDDRV_TimPs	LCDDRV timing_Ps control	R/W	0x10
LCDDRV_TimSplr	LCDDRV timing_Splr control	R/W	0x14
LCDDRV_TimRev	LCDDRV timing_Rev control	R/W	0x18
LCDDRV_TimSps	LCDDRV timing SPS control	R/W	0x1C
LCDDRV_TimPs2Ps3	LCDDRV timing PS2PS3 control	R/W	0x20
LCDDRV_TimPs2	LCDDRV timing PS2 control	R/W	0x24
LCDDRV_TimAsc1	LCDDRV timing ASC1 control	R/W	0x28
LCDDRV_TimAsc2	LCDDRV timing ASC2 control	R/W	0x2C
LCDDRV_TimSsc1	LCDDRV timing SSC1 control	R/W	0x30
LCDDRV_TimSsc2	LCDDRV Timing SSC2 control	R/W	0x34

Table 4–7. LCD Driver Functional Control Register 0 (LCDDRV_FNCTL0)

Base Address = 0xFFFC:2000, Offset = 0x00			
Bit	**Name**	**Description**	**Reset**
15	Reserved	Reserved – set to zero	0
14	PixG	Pixel clock gating: 0: Pixel clock toggles always. 1: Pixel clock only toggles when there is valid data to display.	0
13	IPS3	Invert power saving signal 3: 0: PS3 signal pin is active high. 1: PS3 signal pin is active low.	0
12	IPS2	Invert power saving signal 2: 0: PS2 signal pin is active high. 1: PS2 signal pin is active low.	0

Table 4–7. LCD Driver Functional Control Register 0 (LCDDRV_FNCTL0) (Continued)

Base Address = 0xFFFC:2000, Offset = 0x00			
Bit	**Name**	**Description**	**Reset**
11	ICL	Invert CLS signal: 0: CLS signal pin is active high. 1: CLS signal pin is active low.	0
10	ISLR	Invert SPL/SPR signal: 0: SPL/SPR signal pin is active high. 1: SPL/SPR signal pin is active low.	0
9	IRV	Invert REV signal: 0: REV signal pin is active high. 1: REV signal pin is active low.	0
8	IPS	Invert power saving signal: 0: PS signal pin is active high. 1: PS signal pin is active low.	0
7	IOE	Invert output enable signal: 0: LCD_AC_ENAB pin is active high. 1: LCD_AC_ENAB pin is active low. The IOE bit is used to select the active polarity enable of the pin LCD_ac_enab. This pin is used to indicate to the LCD panel when valid display data is available. Data is driven onto the LCD data lines at the programmable edge of the LCD_ac_enab when it is in its active state	0
6	IPC	Invert panel clock signal: 0: Data is driven on the LCD data lines on the rising edge of the DCLK(Dot clock) 1: Data is driven on the LCD data lines on the falling edge of the DCLK(Dot Clock) The value programmed in this bit must be the same as the value programmed in the omap lcd controller for proper operation.	0
5	IhS	Invert horizontal synchronization signal: 0: LP is active high and inactive low. 1: LP is active low and inactive high.	0
4	IVS	Invert vertical synchronization: 0: SPS is active high and inactive low. 1: SPS is active low and inactive high.	0

Table 4–7. LCD Driver Functional Control Register 0 (LCDDRV_FNCTL0) (Continued)

Base Address = 0xFFFC:2000, Offset = 0x00			
Bit	**Name**	**Description**	**Reset**
3	PSND	During non-display period PS3 can be output on the PS signal by setting this bit: 0: The PS3 signal is disable and PS1 or PS2 (this is again controlled by bit 2 PSD) is output on the PS signal. 1: PS3 is output on the PS signal.	0
2	PSD	During display period, PS1 or PS2 can be output on the PS signal, depending on the power saving requirement: 0: PS1 is output on the PS signal. 1: PS2 is output on the PS signal.	0
1	PIB	Pixel blanking bit is used to blank out the data bits by setting all data bits to zero during the non-display periods on the screen. This bit only has effect in TFT or HR-TFT mode Non-display period = The entire vertical sync period The entire vertical back porch period The entire vertical front porch period The horizontal sync period The horizontal front porch period The horizontal back porch period 0: Data bits are not forced to zero. 1: Data bits are forced to zero during non-display period. BlankPixel: TFT mode/HR-TFT mode **&** SW bit is set to blank during inactive **&** LCD.AC is inactive (low).	0
0	HRM	HR-TFT mode bypass bit is used to put the LCD controller in the HR-TFT Mode. This is the master enable bit for HR-TFT mode. 0: All the control signals of the HR-TFT sub-block are inactive. The HR-TFT sub-block is bypassed. 1: All the control signals are active for HR-TFT mode. This bit must be set to 1 to enable HR-TFT functionality. In addition the LCD controller must also be in TFT mode by setting LcdTFT=1. All other registers must be programmed first and then the control register must be programmed to enter into HR-TFT mode	0

Table 4–8. LCD Driver Functional Control Register 1 (LCDDRV_FNCTL1)

Base Address = 0xFFFC:2000, Offset = 0x04			
Bit	**Name**	**Description**	**Reset**
15:6	Reserved	Reserved – set to zero	0
5	ISSC	Invert SSC For the source signal control SSC 0: SSC is active high and inactive low. 1: SSC is active low and inactive high.	
4	IASC	Invert ASC For the analog switch control signal ASC 0: ASC is active high and inactive low. 1: ASC is active low and inactive high.	0
3:2	///	Reserved – set to zero	0
1	PixPwdn	Pixel clock power down Pixel clock gating strategy 0: The input pixel clock is free running to the pixel clock domain logic. This is default. 1: The input pixel clock is shut off to the logic running on the pixel clock domain. This bit has to be set only when running the LCD_DRIVER module in bypass mode(LCDDRV_FnCtl0(0) = 0), as in this mode the hr-tft signals are rendered inactive. If LCDDRV_FnCtl0(0) = 1, that is, while running in non-bypass (HR-TFT mode) , then this bit has no effect and the input pixel clock is free running.	0
0	Auld	Auto-idle Internal OCP clock gating strategy 0: OCP clock is free running. 1: Automatic OCP clock gating strategy is applied, based on the OCP interface activity. When this bit is set to 1, then the autogating feature is active in the OCP clock domain. Whenever there is no activity on the MCMD signal of the OCP interface (that is, IDLE mode), the OCP clock is shut off to the entire OCP clock domain. The clock is only active when there is any command other than IDLE on the MCMD signal	0

Table 4–9. LCD Driver Timing_Lp Control Register (LCDDRV_TIMLP)

Base Address = 0xFFFC:2000, Offset = 0x08			
Bit	**Name**	**Description**	**Reset**
15:0	LPDEL	LP delay Controls the delay (number of DCLK periods) from HSYNC to the rising edge of the generated LP. Program with (value required – 1).	h0000

Table 4−10. LCD Driver Timing_Cls Control Register (LCDDRV_TIMCLS)

Base Address = 0xFFFC:2000, Offset = 0x0C			
Bit	**Name**	**Description**	**Reset**
15:7	CLSDEL2	CLS2 delay Controls the delay (number of DCLK) from the rising edge of the SPL/SPR to the falling edge of CLS signal. Program with (value required − 1).	h0000
6:0	CLSDEL1	CLS1 delay Controls the delay (number of DCLK) from the HSYNC to the edge of the generated CLS signal. Program with (value required − 1).	h0000

Table 4−11. LCD Driver Timing_Ps Control Register (LCDDRV_TIMPS)

Base Address = 0xFFFC:2000, Offset = 0x10			
Bit	**Name**	**Description**	**Reset**
15:7	PSDEL2	PS2 delay Controls the delay (number of DCLK) from the rising edge of the SPL/SPR to the rising edge of PS signal. Program with (value required − 1).	h0000
6:0	PSDEL1	PS1 delay Controls the delay (number of DCLK) from the HSYNC to the edge of the generated PS signal. Program with (value required − 1).	h0000

Table 4−12. LCD Driver Timing_Spir Control Register (LCDDRV_TIMSPLR)

Base Address = 0xFFFC:2000, Offset = 0x14			
Bit	**Name**	**Description**	**Reset**
15:0	SPLRDEL	SPL/SPR delay Controls the delay(number of DCLK periods) of the SPL/SPR to indicate the first valid data for the line. Program with (value −1).	h0000

Table 4−13. LCD Driver Timing_Rev Control Register (LCDDRV_TIMREV)

Base Address = 0xFFFC:2000, Offset = 0x18			
Bit	**Name**	**Description**	**Reset**
15:0	REVDEL	REV delay Controls the delay(number of DCLK periods) from HSYNC to the falling edge of the generated REV. Program with (value required − 1).	h0000

Table 4−14. LCD Driver Timing_Sps Control Register (LCDDRV_TIMSPS)

Base Address = 0xFFFC:2000, Offset = 0x1C			
Bit	**Name**	**Description**	**Reset**
15:4	SPSDEL	SPS delay Controls the delay (number of HSYNC) of the SPS pulse from the generation of Vsync signal.program with (value required − 1).	h000
3:0	SPSPW	SPS pulse width Defines the pulse width of the SPS signal in terms of HSYNC pulses (program with value −.1).	h0

Table 4−15. LCD Driver Timing_PS2PS3 Control Register (LCDDRV_TIMPS2PS3)

Base Address = 0xFFFC:2000, Offset = 0x20			
Bit	**Name**	**Description**	**Reset**
15:8	ADEL	ADEL pulse width This value specifies the adel pulse width that is required on ps2 power saving signal. Program with value − 1.	h00
7:0	PS3DEL	PS3 pulse width This value controls the pulse width of the active period of the ps3 signal. Program with value − 1.	h00

Table 4−16. LCD Driver Timing_PS2 Control Register (LCDDRV_TIMPS2)

Base Address = 0xFFFC:2000, Offset = 0x24			
Bit	**Name**	**Description**	**Reset**
15:8	PS2DEL2	PS2DEL2 pulse width This value specifies the ps2del2 pulse width that is required on ps2 power saving signal. Program with value −1.	h00
7:0	PS3DEL	PS2DEL1 pulse width This value specifies the ps2del1 pulse width that is required on ps2 power saving signal. Program with value − 1.	h00

Table 4−17. LCD Driver Timing_ASC1 Control Register (LCDDRV_TIMASC1)

Base Address = 0xFFFC:2000, Offset = 0x28			
Bit	**Name**	**Description**	**Reset**
15:4	Tascdel	TASCDEL delay Controls the delay (number of pixel clocks) of the first ASC pulse from the generation of REV signal with (value required − 1).	h000
3:0	Twasc	Pulse width of the ASC signal This value specifies the TWASCD pulse width that is required on ASC signal. Program with value − 1.	h0

Table 4−18. LCD Driver Timing_ASC2 Control Register (LCDDRV_TIMASC2)

Base Address = 0xFFFC:2000, Offset = 0x2C			
Bit	**Name**	**Description**	**Reset**
15:8	Tpasc2	Pulse period 2of ASC signal This value specifies the pulse period 2 that is required on ASC signal. Program with value −1.	h00
7:0	Tpasc1	Pulse period 1 of ASC signal This value specifies the pulse period 1 that is required on ASC signal. Program with value −1.	h00

Table 4−19. LCD Driver Timing_SSC1 Control Register (LCDDRV_TIMSSC1)

Base Address = 0xFFFC:2000, Offset = 0x30			
Bit	**Name**	**Description**	**Reset**
15:4	Tsscdel	Tsscdel delay Controls the delay(number of pixel clocks) of the first SSC pulse from the generation of REV signal with(value required − 1).	h000
3:0	Twssc	Pulse width of the SSC signal This value specifies the Twssc pulse width that is required on SSC signal. Program with value −1.	h0

Table 4−20. LCD Driver Timing_SSC2 Control Register (LCDDRV_TIMSSC2)

Base Address = 0xFFFC:2000, Offset = 0x34			
Bit	**Name**	**Description**	**Reset**
15:0	Tpssc	Pulse period of SSC signal This value specifies the pulse period that is required on SSC signal. Program with value −1.	h00

4.5 Programming Model

4.5.1 OMAP LCD Controller Settings

The OMAP LCD controller settings are as follows:

- When the LCD driver module is operated in bypass mode (LCDDRV_FnCtl(0) = 0), there are no restrictions placed on the register settings of the OMAP LCD controller. The signals coming out of the OMAP LCD controller are directly passed out of the LCD driver module without any change.
- When the LCD driver module is operated in non-bypass mode (LCDDRV_FnCtl(0) = 1), there are certain restrictions placed on certain registers of the OMAP LCD Controller. Table 4–21 and Table 4–22 list the registers of the OMAP LCD controller that must be programmed for the LCD driver module to function correctly. The registers (or register bits) that are not listed in the table can be programmed as per the OMAP LCD controller functional spec, since there are no restrictions placed on it.

Table 4–21. OMAP850 LCD Control Register (LCDCONTROL)

Base Address = 0xFFFE:C000, Offset = 0H00			
Bit	**Name**	**Description**	**Value to be Programmed**
7	LcdTFT	LCD TFT 1: Active or TFT display operation enable to support continuous pixel clock, output enable, VSYNC, and HSYNC	1
11	PXL_GATED	Pixel gated (For TFT mode only) 0: Pixel clock toggles always	0

Table 4–22. OMAP850 LCD Timing 2 Register (LCDTIMING2)

Base Address = 0xFFFE:C000, Offset = 0H0C			
Bit	**Name**	**Description**	**Value to be Programmed**
20	IVS	Invert VSYNC 0: The input pin VSYNC from the OMAP LCD controller is pin is active high and inactive low.	0
21	IHS	Invert HSYNC 0: The input pin HSYNC from the OMAP LCD controller pin is active high and inactive low.	0
23	IEO	Invert output enable 0: AC_BIAS input pin from the omap lcd controller acts as output enable and is active high during active pixel data display. Data is driven on the programmed pixel clock edge(bit 22).	0

Table 4–22. OMAP850 LCD Timing 2 Register (LCDTIMING2) (Continued)

Base Address = 0xFFFE:C000, Offset = 0H0C			
24	RF	Program HSYNC/VSYNC RISE OR FALL 1: HSYNC and VSYNC input pins are driven on rising edge of pixel clock (bit 25 must be set to 1).	1
25	ON_OFF	HSYNC/VSYNC pixel clock control on/off 1: HSYNC and VSYNC input pins are driven according to bit 24.	1

After the OMAP LCD controller registers are programmed incorporating the restrictions listed in Table 4–21 and Table 4–22, you can program the registers of the LCD driver.

4.5.2 Power Management

There are two features of power management in the LCD driver module:

- OCP autogating is controlled by bit LCDDRV_FnCtl1(0). When enabled by setting this bit to 1, the activity on the MCMD signal of the OCP interface is monitored. If an IDLE code is decoded, then the ocp_clk to the module is shut down. For any other mode, the OCP_CLK is enabled and normal transactions take place. With this feature enabled there are no limitations on writing or reading from the LCD driver module.
- Pixel clock shut down is controlled by bit LCDDRV_FnCtl1(1). This bit determines if the PIXEL_CLK to the block logic can be shut down. This feature is useful when operating the LCD driver in bypass mode, when the functionality of the HR-TFT is not required. This bit however has no effect when operating in non-bypass or HR-TFT mode.

4.5.2.1 Deep/Big Sleep

The LCD driver module runs on the traffic controller clock domain. In deep sleep mode, the MPU and the DPLLs and the external clock references are stopped. The LCD pixel clock can be shut off by writing 0 to the corresponding bits of the ARM_IDLECT2 register, and then the MPU can be put into idle mode leaving the device in a low-power mode consuming the lowest power possible.

Chapter 5

Configuration Registers

This chapter describes the configuration registers of the OMAP850 module.

Topic **Page**

5.1 OMAP730 Configuration Registers 5-2

5.1 OMAP850 Configuration Registers

Table 5−1 lists the OMAP850 configuration registers. Table 5−2 through Table 5−55 describe the individual register bits.

Table 5−1. OMAP850 Configuration Registers

Base Address = 0xFFFE:1000				
Name	**Description**	**Bits**	**R/W**	**Offset**
PERSEUS2_MPU_DEV_ID	Device identification on MPU side	32	R	0x00
PERSEUS2_GSM_DEV_ID0	Device identification 0 on GSM side	16	R	0x00
PERSEUS2_GSM_DEV_ID1	Device identification 1 on GSM side	16	R	0x02
DSP_CONF	Software compatibility with EDGE	16	R/W	0x04
PERSEUS2_MPU_DIE_ID0	OMAP850 die identification 0	32	R	0x08
GSM_ASIC_CONF	Compatibility with TBB2100	16	R/W	0x08
PERSEUS2_MPU_DIE_ID1	OMAP850 die identification 1	32	R	0x0C
PERSEUS2_MODE1	OMAP850 mode configuration 1	32	R/W	0x10
PERSEUS2_GSM_DIE_ID0	OMAP850 die identification 0 on GSM side	16	R	0x10
PERSEUS2_GSM_DIE_ID1	OMAP850 die identification 1 on GSM side	16	R	0x12
PERSEUS25_MODE2	OMAP850 mode configuration 2	32	R/W	0x14
PERSEUS2_GSM_DIE_ID2	OMAP850 die identification 2 on GSM side	16	R	0x14
PERSEUS2_GSM_DIE_ID3	OMAP850 die identification 3 on GSM side	16	R	0x16
PERSEUS2_ANALOG_ CELLS_CONF	OMAP850 analog cells configuration	32	R/W	0x18
SECCTRL	Secure control	32	R/W	0x1C
SPARE1	ECO spare 1	32	R/W	0x20
SPARE2	ECO spare 2	32	R/W	0x24
GSM_PBG_IRQ	EDGE − GSM domain	16	R/W	0x28
DMA_REQ_CONF	DMA mode configuration	32	R/W	0x30
PE_CONF_NO_DUAL	EDGE − MPU domain	32	R/W	0x60
PERSEUS2_IO_CONF0	OMAP850 input/output configuration 0	32	R/W	0x70
PERSEUS2_IO_CONF1	OMAP850 input/output configuration 1	32	R/W	0x74
PERSEUS2_IO_CONF2	OMAP850 input/output configuration 2	32	R/W	0x78
PERSEUS2_IO_CONF3	OMAP850 input/output configuration 3	32	R/W	0x7C
PERSEUS2_IO_CONF4	OMAP850 input/output configuration 4	32	R/W	0x80
PERSEUS2_IO_CONF5	OMAP850 input/output configuration 5	32	R/W	0x84

Table 5−1. OMAP850 Configuration Registers (Continued)

Base Address = 0xFFFE:1000				
PERSEUS2_IO_CONF6	OMAP850 input/output configuration 6	32	R/W	0x88
PERSEUS2_IO_CONF7	OMAP850 input/output configuration 7	32	R/W	0x8C
PERSEUS2_IO_CONF8	OMAP850 input/output configuration 8	32	R/W	0x90
PERSEUS2_IO_CONF9	OMAP850 input/output configuration 9	32	R/W	0x94
PERSEUS2_IO_CONF10	OMAP850 input/output configuration 10	32	R/W	0x98
PERSEUS2_IO_CONF11	OMAP850 input/output configuration 11	32	R/W	0x9C
PERSEUS2_IO_CONF12	OMAP850 input/output configuration 12	32	R/W	0xA0
PERSEUS2_IO_CONF13	OMAP850 input/output configuration 13	32	R/W	0xA4
PERSEUS_PCC_CONF_REG	48-MHz Input Control	32	R/W	0xB4
BIST_STATUS_INTERNAL	BIST fail go	32	R	0xB8
BIST_CONTROL	BIST settings control	32	R/W	0xC0
BOOT_ROM_REG	Boot procedure	32	R/W	0xC4
PRODUCTION_ID_REG	Secure chip	32	R/W	0xC8
BIST_SECROM_ SIGNATURE1_INTERNAL	Secure ROM signature 1	32	R/W	0xD0
BIST_SECROM_ SIGNATURE2_INTERNAL	Secure ROM signature 2	32	R/W	0xD4
BIST_CONTROL_2	BIST settings control 2	32	R/W	0xD8
DEBUG1	Debug signal selection 1	32	R/W	0xE0
DEBUG2	Debug signal selection 2	32	R/W	0xE4
DEBUG_DMA_IRQ	DMA and IRQ selection	32	R/W	0xE8
PERSEUS25_MODE	SW compatibility purpose	32	R/W	0xEC
PERSEUS25_FADD_IOCONF1	Configuration register for the addresses 8 to 1	32	R/W	0xF0
PERSEUS25_FADD_IOCONF2	Configuration register for the addresses 15 to 9	32	R/W	0xF4
PERSEUS25_DAGON_MODE	Control of TWC310 module	32	R/W	0xF8
PERSEUS25_DAGON_IO_ CONF0	OMAP850 shared IOs configuration 0	32	R/W	0xFC
PERSEUS25_DAGON_IO_ CONF1	OMAP850 shared IOs configuration 1	32	R/W	0x100
DEBUG_DAGON_0	Selection of dual debug signals for TWC310	32	R/W	0x104
DEBUG_DAGON_1	Selection of dual debug signals for TWC310	32	R/W	0x108
PCONF_DAGON_JTAG_CTRL	OMAP850 shared IOs configuration	32	R/W	0x10C

Table 5−2. Device Identification on MPU−S side Register (PERSEUS2_MPU_DEV_ID)

Base Address = 0xFFFE:1000, Offset = 0x00				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:21	MANUFACTURER_IDENTITY	Texas Instruments IEEE ID=0x17	R	0x17
20	UNUSED	Always 1	R	0x1
19:16	MPU_VERSION	Chip version number identical to GSM_VERSION	R	0x1
15:0	MPU_PART_NUMBER	Device part number in JTAG format identical to GSM_PART_NUMBER	R	0xB62C

Table 5−3. Device Identification 0 (LSBs) on GSM-S side Register (PERSEUS2_GSM_DEV_ID0)

Base Address = 0xFFFE:1000, Offset = 0x00				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	GSM_PART_NUMBER	Device part number in JTAG format Identical to MPU_PART_NUMBER	R	Unknown

Table 5−4. Device Identification 1 (LSBs) on GSM−S side Register (PERSEUS2_GSM_DEV_ID1)

Base Address = 0xFFFE:1000, Offset = 0x02				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:4	RESERVED	RESERVED	R	0X0
3:0	VERSION_GSM	Chip version number: identical to VERSION_MPU	R	0x1

Table 5−5. Software Compatibility with EDGE Register (DSP_CONF)

Base Address = 0xFFFE:1000, Offset = 0x04				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:8	UNUSED	Unused	R/W	0x0
7	TPU_FRAME_INTERRUPT	0: TPU frame IRQ disable 1: TPU frame IRQ enable	R/W	0x0
6:0	UNUSED	Unused	R/W	0x0

Table 5−6. OMAP850 Die Identification 0 on MPU−S Side Register (PERSEUS2_MPU_DIE_ID0)

Base Address = 0xFFFE:1000, Offset = 0x08				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	PERSEUS2_MPU_DIE_ID0	Perseus2 DIE_ID0	R	Unknown

Table 5−7. Software Compatibility with TBB2100 Register (GSM_ASIC_CONF)

Base Address = 0xFFFE:1000, Offset = 0x08				
Bit	**Name**	**Function**	**R/W**	**Reset**
15	UNUSED	Unused	R/W	0x0
14	SPI_CLK_POL	0: SPI config clock RX (falling edge). 1: SPI config clock RX (rising edge).	R/W	0x0
13	RIF_CLK_POL	0: RIF config clock RX (falling edge). 1: RIF config clock RX (rising edge).	R/W	0x0
12:0	UNUSED	Unused	R/W	0x0

Table 5−8. OMAP850 Die Identification 1 on MPU−S Side Register (PERSEUS2_MPU_DIE_ID1)

Base Address = 0xFFFE:1000, Offset = 0x0C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	PERSEUS2_MPU_DIE_ID1	Perseus2 DIE identification number 1	R	Unknown

Table 5−9. OMAP850 Mode Configuration 1 Register (PERSEUS2_MODE1)

Base Address = 0xFFFE:1000, Offset = 0x10				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:30	LED_OUT_GSM	00: LPG1/LPG2 disabled 01: LPG1 enabled on nper_gsm_mlpg1 pin 10: LPG2 enabled on mpu_nboot_mlpg2 pin 11: LPG1 and LPG2 enabled	R/W	0X00
29:28	LED_OUT_MPU	00: LPG1/LPG2 disabled 01: LPG1 enabled on NPER_GSM_MLPG1 pin 10: LPG2 enabled on MPU_NBOOT_MLPG2 pin 11: LPG1 and LPG2 enabled	R/W	0X00
27	MPU_SYREN_SPI_CLK_POL	0: MPU_SPI config clock RX (Fallingedge) 1: MPU_SPI config clock RX (Risingedge)	R/W	0X0
26	USB_TRANSCEIVER_SEL	0: External USB transceiver 1: On−chip USB transceiver	R/W	0X0
25	USB_VBUS_CTRL	When USB_VBUS_MODE = 1 : drive VBUS signal of USB module 0: USB BUS is not plugged. 1: USB BUS is plugged	R/W	0X0
24:23	USB_VBUS_MODE	00: DUAL_MODE pin USB_VBUS drives the power line of USB module 01: The register USB_VBUS_CTRL drives the power line 10: Power line driven by tactical cell UIS480	R/W	0X1

Table 5−9. OMAP850 Mode Configuration 1 Register (PERSEUS2_MODE1) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x10				
Bit	**Name**	**Function**	**R/W**	**Reset**
22:19	EXT_IO_CTRL	0: Control output register, 1: Control input buffer EXT_IO_CTRL(5) => EXT_IO_3 EXT_IO_CTRL(4) => EXT_IO_2 EXT_IO_CTRL(3) => EXT_IO_1 EXT_IO_CTRL(2) => EXT_IO_0	R/W	0X0
18	EXT_IO_CS_SEL	0: CS2 1: CS3	R/W	0X0
17	EXT_IO_EN	0: Disable EXT_IO‘ 1: Enable: CS3 (or CS2) disable when Add = FFFFF8 − FFFFFE and Ext_io enable EXT_IO_3 = Add 0xFFFFFE EXT_IO_2 = Add 0xFFFFFC EXT_IO_1 = Add 0xFFFFFA EXT_IO_0 = Add 0xFFFFF8	R/W	0x0
16	PIARMTDBGEN	Setting MPUTDMI TDBGEN in functionnal mode.	R/W	0x0
15	USB_TRANSCEIVER_SPEED	To internal transceiver : 0: Low speed, 1: High speed.	R/W	0x0
14	MPU_SPI1_CLK_POL	0: MPU_SPI config clock RX (Fallingedge) 1: MPU_SPI config clock RX (Rising edge)	R/W	0x0
13	TC_LRU_SEL	0: TC uses Fixed priority algorithm 1: TC uses LRU (Least Recently Use) mode priority algorithm	R/W	0x1
12	NANDF_CTRL_ENA	0: NAND flash controller disable 1: NAND flash controller enable	R/W	0x0
11:10	USB_PWRDN	D+ D− pull down control 00:pull down enable on D+ D− 01:PWRDN1 active, pull down on D+ 10:PWRDN2 active, pull down on D− 11:pull down disable on D+ D−	R/W	0x0
9	EMULATION_CONFIG	0: MPU /GSM has separate signals /(MPU emu signals on nemu0/1 pins Dsp/Arm7 emu signals on nemu0/1 dvpt) 1: Shared MPU−GSM EMU signals	R/W	0x0
8	PIARMTDBGRQ	Setting MPUTDMI TDBGRQ in functionnal mode.	R/W	0x0
7	ON_OFF	0: Signal ON_OFF (GSM Power Mngt) from pin ON_OFF 1: Signal ON_OFF (GSM Power Mngt) from ULPD	R/W	0x0

Table 5–9. OMAP850 Mode Configuration 1 Register (PERSEUS2_MODE1) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x10				
Bit	**Name**	**Function**	**R/W**	**Reset**
6	MCBSP2_CLKS_SEL	0: Internal oscillator from PCC (13/48 Mhz) 1: External clock from PAD	R/W	0x00
5	MPU_SPI2_CLK_POL	0: MPU_SPI2 config clock RX (Fallingedge) 1: MPU_SPI2 config clock RX (Rising edge)	R/W	0x0
4	MCBSP1_CLKS_SEL	Input MCLK of EAC and input CLKS of MCBSP : 0: Internal oscilator, from PCC (13/48 Mhz) 1: External CODEC CLOCK, from pad.	R/W	0X0
3:2	INTERNAL_EAC_BT_ AUSPI_SOURC	Selected source to EAC Bt AuSPI Port Module 00: Pins BT AuSPI : SCLK, SDI, FSYNC 01: Pins MPU_SPI : MPU_SCLK, MPU_SDI, MPU_SEN1 10: Reserved 11: gated 0	R/W	0x3
1:0	INTERNAL_GSM_ VOICE_SOURCE	Selected source to Lead2 vspi (gsm voice), already slave 00: Pins BT AuSPI : SCLK, SDI, FSYNC 01: Internal EAC Modem AuSPI 10: Internal VSPI Modules, 11: Nothing, gated 0	R/W	0x3

Table 5–10. OMAP850 Die Identification 0 on GSM-S Side Register (PERSEUS2_GSM_DIE_ID0)

Base Address = 0xFFFE:1000, Offset = 0x10				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	GSM_DIE_ID_0	DIE ID 0	R	Unknown

Table 5–11. OMAP850 Die Identification 1 on GSM-S Side Register (PERSEUS2_GSM_DIE_ID1)

Base Address = 0xFFFE:1000, Offset = 0x12				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	GSM_DIE_ID_1	DIE ID 1	R	Unknown

Table 5–12. OMAP850 Mode Configuration 2 Register (PERSEUS25_MODE2)

Base Address = 0xFFFE:1000, Offset = 0x14				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:18	UNUSED	UNUSED bits	R/W	0x0
17	RESERVED	Reserved	R/W	0x1

Table 5−12. OMAP850 Mode Configuration 2 Register (PERSEUS25_MODE2) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x14				
Bit	**Name**	**Function**	**R/W**	**Reset**
16	NFCS3HL_16_32M	0: NFCCS0L and NFCS0H for 16 MB flash 1: For 32-MB flash	R/W	0x0
15	VLYNQ_CLKOUT_EN	VLYNQ clock enable	R/W	0x1
14	V2O_CLK_EN	OCP clock enable	R/W	0x1
13	VOLTAGE_FLASH	Dual-voltage control for flash domain: 0: 1.65 V min, 1.8 V nom, 1.95 V max 1: 2.5 V min, 2.75 V nom, 3.0 V max	R/W	0x1
12	VOLTAGE_SDRAM	Dual-voltage control for SDRAM domain 0: 1.65 V min, 1.8 V nom, 1.95 V max 1: 2.5 V min, 2.75 V nom, 3.0 V max	R/W	0x1
11	RNG_TESTOSC	Used to get test feature (1: Ring oscillator output test)	R/W	0x0
10	RNG_SELECTOSC	Ring oscillator 1 or 2 selection (0: oscillator 1 / 1: oscillator 2)	R/W	0x0
9:8	DLL_SELECT	Select DLL/DCDL: When x0 PSTART, PFBK and DCB[7:0] are selected from URD_DLL. When x1 PSTART, PFBK and DCB[7:0] are selected from WRQ_DLL. When 00 PMT_DSI and PMT_DSO are selected from URD_DCDL When 01 PMT_DSI and PMT_DSO are selected from WRQ_DCDL When 1x PMT_DSI and PMT_DSO are selected from LRD_DCDL	R/W	0x00
7	JTAG_EN	0: Disable OMAP JTAG access 1: Enable OMAP JTAG access	R/W	0x0
6	LCD_PSEUDO_18BIT	0 (reset value): 16-bit mode, the pseudo red and blue 6th bits are set to 0. Used to create a 64K color display. 1: Pseudo 18-bit mode, the pseudo red and blue 6th bits are a copy of 6th bit (LSB) green.	R/W	0x0
5	SD_REQ_BYPASS	0: SDRAM_REQ = not GPIO4 and ICR_GSM_RST 1: SDRAM_REQ = not GPIO4	R/W	0x0

Table 5−12. OMAP850 Mode Configuration 2 Register (PERSEUS25_MODE2) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x14				
Bit	**Name**	**Function**	**R/W**	**Reset**
4:3	TAP_CONF	Control the link of all perseus2 TAPs: 00: MPU mono emulation 01: MPU/DSP bi-emulation 10: MPU/arm7/lead2 tri-emulation 11: Not used	R/W	0x2
2	OCP_INTERC_CGM_ENABLE	Enable gated clock feature	R/W	0x1
1	OCP_INTERC_SW_RESET	Software reset input signal (active high)	R/W	0x0
0	VLYNQ_DEFAULT_CLKDIR	The serial clock may be sourced by the internal VBUS clock (clkdir=1) or by an external clock (cllkdir=0). The default value of the clkdir bit is set during the reset with the value on the VLYNQ_DEFAULT_CLKDIR input signal.	R/W	0x0

Table 5−13. OMAP850 Die Identification 2 on GSM-S Side Register (PERSEUS2_GSM_DIE_ID2)

Base Address = 0xFFFE:1000, Offset = 0x14				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	GSM_DIE_ID_2	DIE ID 2	R	Unknown

Table 5−14. OMAP850 Die Identification 3 on GSM-S side Register (PERSEUS2_GSM_DIE_ID3)

Base Address = 0xFFFE:1000, Offset = 0x16				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	GSM_DIE_ID_3	DIE ID 3	R	Unknown

Table 5−15. Analog Cells Configuration Register (PERSEUS2_ANALOG_CELLS_CONF)

Base Address = 0xFFFE:1000, Offset = 0x18				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:6	RESERVED	RESERVED	R/W	0x0
5	SLICER_PWRSEL	When 1 the slicer is in high-power application and when 0 the slicer is in low-power application. This bit selects either the low-power mode (PWRSEL low) or the high-performance mode (PWRSEL high). The high-performance mode allows better duty cycle than in low-power mode. The power cost is 300 μA. In high performance mode by default.	R/W	0x1

Table 5−15. Analog Cells Configuration Register (PERSEUS2_ANALOG_CELLS_CONF) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x18				
Bit	**Name**	**Function**	**R/W**	**Reset**
4	SLICER_PWRDN	When selected (equal 1) the cell does not consume any current if the bypass mode is not active	R/W	0x0
3:0	CONTROL_ANALOG_ SWITCHES	32k oscillator current gain control analog switches from SW1 to SW3 (see TI OS11V1 cell specification)	R/W	0x8

Table 5−16. Secure Control Register (SECCTRL)

Base Address = 0xFFFE:1000, Offset = 0x1C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:17	RESERVED	RESERVED	R/W	0x0
16:12	UNUSED	UNUSED	R/W	0x0
11	CONF_RNG_EN	This is the RNG module access control register. This bit is programmable only in secure mode 0 RNG module access in non−secure mode and secure mode is enabled. 1: RNG module access in secure mode only is enabled. Reset: 0: Reset condition for a normal production device. 1: Reset condition for an emulation device. This register can be configured as required and does not have a one time only configuration restriction for either emulation devices or normal production devices.	R/W	0x1
10	CONF_DES_EN	This is the DES/3DES module access control register. This bit is programmable only in secure mode 0: DES/3DES module access in non−secure mode and secure mode is enabled. 1: DES/3DES module access in secure mode only is enabled. Reset: 0: Reset condition for a normal production device. 1: Reset condition for an emulation device. This register can be configured as required and does not have a one time only configuration restriction for either emulation devices or normal production devices.	R/W	0x1

Table 5−16. Secure Control Register (SECCTRL) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x1C				
Bit	**Name**	**Function**	**R/W**	**Reset**
9	CONF_MUX_CTRL	The register controls access to the registers that configure multiplexing of the device pins and pull-up/pulldown functions. This bit is programmable only in secure mode. 0: The registers FUNC_MUX_CTRL(3 – 12), PULL_DWN_CTRL(0−4), and PU_PD_SEL_(0−4) are accessible at anytime. 1: The registers FUNC_MUX_CTRL(3 – 12), PULL_DWN_CTRL(0−4), and PU_PD_SEL_(0−4) are accessible only while the device is in secure mode. Reset: 0: Reset condition for a normal production device. 1: Reset condition for an emulation device. This register can be configured as required and does not have a one time only configuration restriction for either emulation devices or normal production devices.	R/W	0x0
8	CONF_SHA_EN	Normal production device: R/OTC RandD/SW development device: R SHA−1 module access control register This bit is programmable only in secure mode 1: SHA−1 module access in secure mode only is enabled 0: SHA−1 module access in non−secure and secure mode is enabled note that DMA access to SHA1 is dependant on CONF_SHA_EN and on the register DMA_BLOCK bit in OMAP3.2 Reset: always 1	R/W	0x1
7	NORMAL_EMU_MODE_ INTERNAL	This bit has information about security type: This bit is programmable only in secure mode 0: Security type is normal secure 1: Security type is debug secure This bit is one−time−programmable for normal device and read−only for emulation device. It's reset value is 0 for both normal and emulation device type. Normal production device: R/OTC RandD/SW development device: R	R	0x0

Table 5−16. Secure Control Register (SECCTRL) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x1C				
Bit	**Name**	**Function**	**R/W**	**Reset**
6	CONF_JTAG_EN	Function: MPU JTAG enable control register This bit is programmable only in secure mode 1: MPU JTAG enabled (the functionality of the MPU JTAG is not altered) 0: MPU JTAG disabled Reset: 0: Normal production device 1: RandD/SW development device This register can be configured one time only for a normal production device. It can't be configured at all for an emulation(RandD) device. Normal production device: R/OTC RandD/SW development device: R	R	0x0
5	CONF_ETM_EN	Function: ETM enable control register This bit is programmable only in secure mode 1: Trace is not effected 0: Trace is disabled Reset: 0: Reset condition for a normal production device 1: Reset condition for an RandD/SW development device This register can be configured one time only for a normal production device. It can't be configured at all for an emulation device. Normal production device: R/OTC RandD/SW development device: R	R	0x0
4	CONF_CKEY_ACC	Function: CKEY access control register This bit is programmable only in secure mode 1: e−fuse access to field C allowed 0: e−fuse access to field C forbidden Reset: 0: Reset condition for a normal production device 1: Reset condition for a an emulation(RandD) device This register can be configured one time only for both normal and emulation devices.	R/W1	0x1

Table 5−16. Secure Control Register (SECCTRL) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x1C				
Bit	**Name**	**Function**	**R/W**	**Reset**
3	CONF_RKEY_ACC	Function: Rkey access control register This bit is programmable only in secure mode 1: Salt access allowed 0: Salt access forbidden Reset: 1: Reset condition for a normal production device 0: Reset condition for an emulation (RandD) device This register can be configured one time only for a normal production device. It can't be configured at all for an emulation device. Normal production device : R/OTC RandD/SW development device: R	R	0x1
2	CONF_ICE_EN	Function: MCU emulation enable control register This bit is programmable only in secure mode 1: MCU debug is not affected. 0: MCU debug is disabled. Reset: 0: Reset condition for a normal production device 1: Reset condition for an emulation(RandD) device This register can be configured one time only for a normal production device. It can't be configured at all for an emulation device. Normal device: R/OTC Development device: R	R	0x0

Table 5−16. Secure Control Register (SECCTRL) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x1C				
Bit	**Name**	**Function**	**R/W**	**Reset**
1	CONF_WDA_EN	Function: Secure Watchdog operation enable control register This bit is programmable only in secure mode 1: If we are in secure mode, OCP access done/ Granted 0: OCP error generated/Rejected Reset: 1:Reset condition for a normal production device 0:Reset condotion for an emulation(RandD) device This register can be configured one time only for both normal and emulation device	R/W1	0x1
0	CONF_WD_ACC	Watchdog access R/OTC register function: Secure watchdog register update access control register This bit is programmable only in secure mode 1: Watchdog timer is frozen. 0: Watchdog timer is running. Reset: 0: Reset condition for a normal production device 1: Reset condition for an emulation(RandD) device This register can be configured one time only for both normal and emulation(RandD) devices.	R/W1	0x1

Table 5−17. Spare 1 Register for ECO (SPARE1)

Base Address = 0xFFFE:1000, Offset = 0x20				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:6	SPARE1	Spare flops for ECO needs	R/W	0x0000000
5:4	CAM_EXCLK_SELECT	Frequency selection of clock output for camera sensor: 00: select clock 48MHz coming from APLL 10: select clock 24MHz coming from APLL x1: select clock ocp_camera coming from TC clock (several divisions avaialable)	R/W	0x0
3	GSM_CLK13M_INPUT	0: CLK13M input of GSM (pi_clkm_slicer) comes divided by 2 duplicated from PCC mechanism 1: CLK13M input of GSM (pi_clkm_slicer) comes from slicer (like P2)	R/W	0x0

Table 5−17. Spare 1 Register for ECO (SPARE1) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x20				
Bit	**Name**	**Function**	**R/W**	**Reset**
2	ARM_BOOT_STATUS	This register contains arm boot mode. Emulation mode: 0: MPU boots from internal ROM 1: MPU boots from external memory Production mode: value read from pin.	R	0xX
1	DISABLE_PULL_SDMC	Disable pullups of SDMC in mode 0	R/W	0x0
0	SPARE1_0	Reserved, must be set to 0	R/W	0x0

Table 5−18. Spare 2 Register for ECO (SPARE2)

Base Address = 0xFFFE:1000, Offset = 0x24				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:6	SPARE2	Spare flops for ECO needs	R/W	0x0
5	RNG_IDLE_MODE	ES1.0: NA ES1.X: RNG idle control: 0: RNG idle disabled 1: RNG idle enabled	R/W	0x1
4	PROTECT_CS3_ENABLE	ES1.0 : NA ES1.X : Flash protect CS3 enable: Enable the mecanisme protection of CS3. Reset Value : The flash CS3 is not protected.	R/W1	0x0
3	PROTECT_CS0_ENABLE	ES1.0: NA ES1.X: Flash protect CS0 enable Enable the mecanisme protection of CS0. Reset Value: The flash CS0 is not protected.	R/W1	0x0

Table 5−18. Spare 2 Register for ECO (SPARE2) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x24				
Bit	**Name**	**Function**	**R/W**	**Reset**
2:1	BLOCK_SIZE	ES1.0: NA ES1.X: Flash protection Size of block protected : 00: 32K 01: 64K 10: 128K 11: 256K	R/W1	0x0
0	P2_IO_CONF0_CONF1_ WR_DIS	ES1.0: NA ES1.X:lock of bits on the PERSEUS2_IO_CONF0 and PERSEUS2_IO_CONF1 register PERSEUS2_IO_CONF0 : 31:29 => Lock of bits 27:25 => Lock of bits 23:21 => Lock of bits 19:17 => Lock of bits 15:13 => Lock of bits 11:9 => Lock of bits 7:5 => Lock of bits 3:1 => Lock of bits PERSEUS2_IO_CONF1 : 3:1 => Lock of bits	R/W1	0x0

Table 5−19.Edge Register (GSM_PBG_IRQ)

Base Address = 0xFFFE:1000, Offset = 0x28				
Bit	**Name**	**Function**	**R/W**	**Reset**
15:0	RESERVED	RESERVED	R/W	0x0

Note: Compliant with TBB2100

Table 5−20.DMA Mode Configuration Register (DMA_REQ_CONF)

Base Address = 0xFFFE:1000, Offset = 0x30				
Bit	**Name**	**Function**	**R/W**	**Reset**
31	RESERVED	RESERVED	R/W	0x0
30:0	DMA_EDGE_EN	DMA request kind 1: Edge 0: Transition	R/W	0x7FFFFFFF

Table 5−21. Pull Enable Control Register (PE_CONF_NO_DUAL)

Base Address = 0xFFFE:1000, Offset = 0x60				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:8	RESERVED	RESERVED	R/W	0X0
7	RESERVED	RESERVED	R/W	0x0

Table 5−21. Pull Enable Control Register (PE_CONF_NO_DUAL) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x60				
Bit	**Name**	**Function**	**R/W**	**Reset**
6	PE_MPU_NRST	Pull enable: 1: Enable 0:Disable	R/W	0x1
5	RESERVED	RESERVED	R/W	0x0
4	PE_NBSCAN	Pull enable: 1: Enable 0:Disable	R/W	0x1
3	PE_TDI	Pull enable: 1: Enable 0: Disable	R/W	0x1
2	PE_TCK	Pull enable: 1: Enable 0:Disable	R/W	0x1
1	PE_NTRST	Pull enable: 1: Enable 0:Disable	R/W	0x1
0	PE_TMS	Pull enable: 1: Enable 0:Disable	R/W	0x1

Note: Special feature for signal with no dual modes. Other pull−enable controls are part of IO_CONF registers

Table 5−22. OMAP850 Shared I/O Configuration 0 Register (PERSEUS2_IO_CONF0)

Base Address = 0xFFFE:1000, Offset = 0x70				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_TPU_TSPEN1	000: TSPEN_1 001: 010: 011: 100: 101: 110: GPIO_8	R/W	0x6
28	PE_TPU_TSPEN1	PE_TPU_TSPEN1 pull enable control	R/W	0x1
27:25	D_TPU_TSPEN0	000: TSPEN_2 001: 010: 011: 100: 101: 110: GPIO_7	R/W	0x6
24	PE_TPU_TSPEN0	PE_TPU_TSPEN0 pull enable control	R/W	0x1

Table 5–22. OMAP850 Shared I/O Configuration 0 Register (PERSEUS2_IO_CONF0) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x70				
Bit	**Name**	**Function**	**R/W**	**Reset**
23:21	D_TPU_TSPACT4	000: TSPACT_4 001: IO_GSM_1 010: 011: 100: 101: 110: GPIO_5	R/W	0x6
20	PE_TPU_TSPACT4	PE_TPU_TSPACt4 pull enable control	R/W	0x1
19:17	D_TPU_TSPACT3	000: TSPACT_3 001: IO_GSM_0 010: 011: 100: 101: 110: GPIO_4	R/W	0x6
16	PE_TPU_TSPACT3	PE_TPU_TSPACT3 pull enable control	R/W	0x1
15:13	D_TPU_TSPACT2	000: TSPACT_2 001: 010: 011: 100: 101: 110: GPIO_3	R/W	0x6
12	PE_TPU_TSPACT2	PE_TPU_TSPACT2 pull enable control	R/W	0x1
11:9	D_TPU_TSPACT1	000: TSPACT_1 001: 010: 011: 100: 101: 110: GPIO_2	R/W	0x6
8	PE_TPU_TSPACT1	PE_TPU_TSPACT1 pull enable control	R/W	0x1
7:5	D_TPU_TSPACT0	000: TSPACT_0 001: 010: 011: 100: 101: 110: GPIO_1	R/W	0x6
4	PE_TPU_TSPACT0	PE_TPU_TSPACT0 pull enable control	R/W	0x1

Table 5–22. OMAP850 Shared I/O Configuration 0 Register (PERSEUS2_IO_CONF0) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x70				
Bit	**Name**	**Function**	**R/W**	**Reset**
3:1	D_TSPCLKX_TSPDO	000: TSPCLKX, TSPDO 001: , 010: , 011: , 100: , 101: , 110: GPIO_0, GPIO_6	R/W	0x6
0	PE_TSPCLKX_TSPDO	PE_SPCLKX_TSPDO pull enable control	R/W	0x1

Table 5–23. OMAP850 Shared I/O Configuration 1 Register (PERSEUS2_IO_CONF1)

Base Address = 0xFFFE:1000, Offset = 0x74				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_RFEN	000: RFEN 001: IO_GSM_2 010: 011: 100: 101: 110: GPIO_19	R/W	0x0
28	PE_RFEN	PE_RFEN pull enable control	R/W	0x1
27:25	D_TCXOEN	000: TCXOEN 001: 010: 011: 100: 101: 110: GPIO_18	R/W	0x0
24	PE_TCXOEN	PE_TCXOEN pull enable control	R/W	0x1
23:21	D_IT_WAKEUP	000: IT_WAKEUP 001: 010: 011: 100: 101: 110:	R/W	0x0
20	PE_IT_WAKEUP	PE_IT_WAKEUP pull enable control	R/W	0x1

Table 5–23. OMAP850 Shared I/O Configuration 1 Register (PERSEUS2_IO_CONF1) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x74				
Bit	**Name**	**Function**	**R/W**	**Reset**
19:17	D_BB_SIM_CD	000: SIM_CD 001: 010: IO_GSM_2 011: MPU_I2C_SCK 100: 101: 110: GPIO_16	R/W	0x6
16	PE_BB_SIM_CD	PE_BB_SIM_CD pull enable control	R/W	0x1
15:13	D_BB_SIM_PWR	000: SIM_PWRCTRL 001: 010: 011: MPU_I2C_SDA 100: 101: 110: GPIO_15 111: HRTFT_SSC	R/W	0x6
12	PE_BB_SIM_PWR	PE_BB_SIM_PWR pull enable control	R/W	0x1
11:9	D_BB_SIM	000: SIM_IO, SIM_CLK, SIM_RST 001: , 010: , 011: , 100: , 101: , 110: GPIO_13, GPIO_14, GPIO_17	R/W	0x6
8	PE_BB_SIM	PE_BB_SIM pull enable control	R/W	0x1
7:5	D_BB_IF	000: BFSR, BDR, BFSX, BDX 001: , , , 010: , , , 011: , , , 100: , , , 101: , , , 110: GPIO_10, GPIN_1, GPIO_11, GPIO_12	R/W	0x6
4	PE_BB_IF	PE_BB_IF pull enable control	R/W	0x1
3:1	D_TPU_TSPEN2	000: TSPEN_0 001: 010: 011: 100: 101: 110: GPIO_9	R/W	0x6
0	PE_TPU_TSPEN2	PE_TPU_TSPEN2 pull enable control	R/W	0x1

Table 5–24. OMAP850 Shared I/O Configuration 2 Register (PERSEUS2_IO_CONF2)

Base Address = 0xFFFE:1000, Offset = 0x78				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_VBUSI	000: USB_VBUSI 001: MPU_UART_RTS1 010: GSM_UART_RTS 011: USB_TXEN 100: GSM_MCSI_TXD 101: MPU_MCSI_TXD 110: GPIO_34	R/W	0x6
28	PE_VBUSI	PE_VBUSI pull enable control	R/W	0x1
27:25	D_PU_EN	000: USB_PU_EN 001: MPU_UART_CTS1 010: GSM_UART_CTS 011: USB_RCV 100: GSM_MCSI_CLK 101: MPU_MCSI_CLK 110: GPIO_33	R/W	0x6
24	PE_PU_EN	PE_PU_EN pull enable control	R/W	0x1
23:21	D_DM	000: USB_DM, USB_DP 001: MPU_UART_TX1, MPU_UART_RX1 010: MPU_UART_TX_IR2, MPU_UART_RX_IR2 011: USB_SEO_VM, USB_TXD_VP 100: USB_SEO, USB_TXD 101: , 110: GPIO_31, GPIO_32	R/W	0x6
20	PE_DM	PE_DM pull enable control	R/W	0x1
19:17	D_SDMC_DAT3	000: SDMC_DAT_3 001: MPU_SPI1_SEN2 010: 011: HRTFT_CLS 100: 101: DG_UART_RTS 110: GPIO_30	R/W	0x6
16	PE_SDMC_DAT3	PE_SDMC_DAT3 pull enable control	R/W	0x1
15:13	D_SDMC_DAT2	000: SDMC_DAT_2 001: MPU_SPI1_SEN1 010: 011: HRTFT_REV 100: 101: DG_UART_CTS 110: GPIO_29	R/W	0x6
12	PE_SDMC_DAT2	PE_SDMC_DAT2 pull enable control	R/W	0x1

Table 5–24. OMAP850 Shared I/O Configuration 2 Register (PERSEUS2_IO_CONF2) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x78				
Bit	**Name**	**Function**	**R/W**	**Reset**
11:9	D_SDMC	000: SDMC_CLK, SDMC_CMD, SDMC_DAT_0, SDMC_DAT_1 001: MPU_SPI1_SCLK, MPU_SPI1_SDO, MPU_SPI1_SDI, MPU_SPI1_SEN0 010: , , , 011: HRTFT_ASC, HRTFT_SSC, HRTFT_SPL, HRTFT_PS 100: , , , 101: , , DG_UART_TX, DG_UART_RX 110: GPIO_25, GPIO_26, GPIO_27, GPIO_28	R/W	0x6
8	PE_SDMC	PE_SDMC pull enable control	R/W	0x1
7:5	D_SYREN_VOICE	000: SCLK, SDO, SDI, FSYNC 001: VCLKRX, VDX, VDR, VFSRX 010: MPU_MCSI_CLK, MPU_MCSI_TXD, MPU_MCSI_RXD, MPU_MCSI_FSYNCH 011: , , , 100: , , , 101: , , , 110: GPIO_22, GPIO_23, GPIN_3, GPIO_24	R/W	0x6
4	PE_SYREN_VOICE	PE_SYREN_VOICE pull enable control	R/W	0x1
3:1	D_SYREN_SPI	000: MCUDI, MCUDO, MCUEN 001: , , 010: , , 011: , , 100: , , 101: , , 110: GPIN_2, GPIO_20, GPIO_21	R/W	0x6
0	PE_SYREN_SPI	PE_SYREN_SPI pull enable control	R/W	0x1

Table 5–25. OMAP850 Shared I/O Configuration 3 Register (PERSEUS2_IO_CONF3)

Base Address = 0xFFFE:1000, Offset = 0x7C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_LCD_PXL_15_12	000: LCD_PIXEL_15, LCD_PIXEL_14, LCD_PIXEL_13, LCD_PIXEL_12 001: GSM_MCSI_FSYNCH, GSM_MCSI_CLK, GSM_MCSI_TXD, GSM_MCSI_RXD 010: MPU_SPI2_SCLK, MPU_SPI2_SDO, MPU_SPI2_SDI, MPU_SPI2_SEN0 011: HRTFT_PIXEL_15, HRTFT_PIXEL_14, HRTFT_PIXEL_13, HRTFT_PIXEL_12 100: , , , 101: DAGON_OBS_15, DAGON_OBS_14, DAGON_OBS_13,DAGON_OBS_12 110: GPIO_44, GPIO_45, GPIO_46, GPIO_47	R/W	0x6
28	PE_LCD_PXL_15_12	PE_LCD_PXL_15_12 pull enable control	R/W	0x1
27:25	D_UART_RTS_CTS	000: MPU_UART_CTS1, MPU_UART_RTS1 001: VLYNQ_RXD1, VLYNQ_TXD1 010: MPU_UART_CTS2, MPU_UART_RTS2 011: GSM_UW_SCLK, GSM_UW_nSCS1 100: MPU_SPI1_SDI, MPU_SPI1_SEN0 101: TSPACT_7, TSPACT_8 110: GPIO_42, GPIO_43	R/W	0x6
24	PE_UART_RTS_CTS	PE_UART_RTS_CTS pull enable control	R/W	0x1
23:21	D_UART_TX_RX	000: MPU_UART_TX1, MPU_UART_RX1 001: VLYNQ_TXD0, VLYNQ_RXD0 010: MPU_UART_TX2, MPU_UART_RX2 011: GSM_UW_SDO, GSM_UW_SDI 100: MPU_SPI1_SCLK, MPU_SPI1_SDO 101: TSPACT_5, TSPACT_6 110: GPIO_40, GPIO_41	R/W	0x6
20	PE_UART_TX_RX	PE_UART_TX_RX pull enable control	R/W	0x1
19:17	D_UART_IRDA_SD	000: MPU_UART_SD2 001: HDQ1W 010: 011: 100: 101: 110: GPIO_39	R/W	0x6
16	PE_UART_IRDA_SD	PE_UART_IRDA_SD pull enable control	R/W	0x1
15:13	D_UART_IRDA_RX	000: MPU_UART_RX_IR2 001: LCD_PIXEL_17 010: 011: 100: GSM_UART_RX 101: 110: GPIO_38	R/W	0x6

Table 5−25. OMAP850 Shared I/O Configuration 3 Register (PERSEUS2_IO_CONF3) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x7C				
Bit	**Name**	**Function**	**R/W**	**Reset**
12	PE_UART_IRDA_RX	PE_UART_IRDA_RX pull enable control	R/W	0x1
11:9	D_UART_IRDA_TX	000: MPU_UART_TX_IR2 001: LCD_PIXEL_16 010: 011: 100: GSM_UART_TX 101: 110: GPIO_37	R/W	0x6
8	PE_UART_IRDA_TX	PE_UART_IRDA_TX pull enable control	R/W	0x1
7:5	D_CRESET	000: CRESET 001: MPU_UART_DSR1 010: GSM_UART_RX 011: USB_VP 100: GSM_MCSI_FSYNCH 101: MPU_MCSI_FSYNCH 110: GPIO_36	R/W	0x6
4	PE_CRESET	PE_CRESET pull enable control	R/W	0x1
3:1	D_MCLK_OUT	000: MCLK_OUT 001: MPU_UART_DCD1 010: GSM_UART_TX 011: USB_VM 100: GSM_MCSI_RXD 101: MPU_MCSI_RXD 110: GPIO_35	R/W	0x6
0	PE_MCLK_OUT	PE_MCLK_OUT pull enable control	R/W	0x1

Table 5−26. OMAP850 Shared I/O Configuration 4 Register (PERSEUS2_IO_CONF4)

Base Address = 0xFFFE:1000, Offset = 0x80				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_EAC_MCLK	000: MCLK 001: VLYNQ_CLK 010: CLKS1 011: TSPACT_10 100: IO_GSM_1 101: CLK13M_IN 110: GPIO_68	R/W	0x6
28	PE_EAC_MCLK	PE_EAC_MCLK pull enable control	R/W	0x1

Table 5–26. OMAP850 Shared I/O Configuration 4 Register (PERSEUS2_IO_CONF4) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x80				
Bit	**Name**	**Function**	**R/W**	**Reset**
27:25	D_EAC_CDI	000: CDI 001: KBC_6 010: DR1 011: TSPACT_9 100: IO_GSM_0 101: 110: GPIO_67	R/W	0x6
24	PE_EAC_CDI	PE_EAC_CDI pull enable control	R/W	0x1
23:21	D_EAC	000: CSYNC, CSCLK, CDO 001: KBR_5, KBR_6, KBC_5 010: FSRX1, CLKRX1, DX1 011: , , 100: , , 101: , , 110: GPIO_64, GPIO_65, GPIO_66	R/W	0x6
20	PE_EAC	PE_EAC pull enable control	R/W	0x1
19:17	D_LCD_VSYNC	000: LCD_VSYNC 001: MPU_UW_nSCS2 010: DR2 011: HRTFT_SPS 100: 101: DAGON_OBS_17 110: GPIO_62	R/W	0x6
16	PE_LCD_VSYNC	PE_LCD_VSYNC pull enable control	R/W	0x1
15:13	D_LCD_UWIRE	000: LCD_PIXEL_0, LCD_PCLK, LCD_HSYNC, LCD_AC 001: MPU_UW_nSCS1, MPU_UW_SCLK, MPU_UW_SDO, MPU_UW_SDI 010: FSRX2, CLKRX2, DX2, CLKS2 011: HRTFT_PIXEL_0, HRTFT_DCLK, HRTFT_LP, HRTFT_ENABLE 100: , , HRTFT CSYNC 101: DAGON_OBS_0, DAGON_OBS_19, DAGON_OBS_18, DAGON_OBS_16 110: GPIO_59, GPIO_60, GPIO_61, GPIO_63	R/W	0x6
12	PE_LCD_UWIRE	PE_LCD_UWIRE pull enable control	R/W	0x1

Table 5–26. OMAP850 Shared I/O Configuration 4 Register (PERSEUS2_IO_CONF4) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x80				
Bit	**Name**	**Function**	**R/W**	**Reset**
11:9	D_LCD_PXL_9_2	000: LCD_PIXEL_9, LCD_PIXEL_8, LCD_PIXEL_7, LCD_PIXEL_6, LCD_PIXEL_5, LCD_PIXEL_4, LCD_PIXEL_3, LCD_PIXEL_2, LCD_PIXEL_1 001: , , , , , , , , 010: , , , , , , , , 011: HRTFT_PIXEL_9, HRTFT_PIXEL_8, HRTFT_PIXEL_7, HRTFT_PIXEL_6, HRTFT_PIXEL_5, HRTFT_PIXEL_4, HRTFT_PIXEL_3, HRTFT_PIXEL_2, HRTFT_PIXEL_1 100: , , , , , , , , 101: DAGON_OBS_9, DAGON_OBS_8, DAGON_OBS_7, DAGON_OBS_6, DAGON_OBS_5, DAGON_OBS_4, DAGON_OBS_3, DAGON_OBS_2, DAGON_OBS_1 110: GPIO_50, GPIO_51, GPIO_52, GPIO_53, GPIO_54, GPIO_55, GPIO_56, GPIO_57, GPIO_58	R/W	0x6
8	PE_LCD_PXL_9_2	PE_LCD_PXL_9_2 pull enable control	R/W	0x1
7:5	D_LCD_PXL_10	000: LCD_PIXEL_10 001: 010: MPU_SPI2_SEN2 011: HRTFT_PIXEL_10 100: 101: DAGON_OBS_10 110: GPIO_49	R/W	0x6
4	PE_LCD_PXL_10	PE_LCD_PXL_10 pull enable control	R/W	0x1
3:1	D_LCD_PXL_11	000: LCD_PIXEL_11 001: 010: MPU_SPI2_SEN1 011: HRTFT_PIXEL_11 100: 101: DAGON_OBS_11 110: GPIO_48	R/W	0x6
0	PE_LCD_PXL_11	PE_LCD_PXL_11 pull enable control	R/W	0x1

Table 5–27. OMAP850 Shared I/O Configuration 5 Register (PERSEUS2_IO_CONF5)

Base Address = 0xFFFE:1000, Offset = 0x84				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_EMIF_FADD22	000: FADD_22 001: 010: 011: MPU_UART_RX_IR2 100: MPU_UART_TX2 101: 110: GPIO_78	R/W	0x0
28	PE_EMIF_FADD22	PE_EMIF_FADD22 pull enable control	R/W	0x1
27:25	D_EMIF_FADD23	000: FADD_23 001: 010: 011: IO_GSM_15 100: MPU_UART_CTS2 101: 110: GPIO_77	R/W	0x0
24	PE_EMIF_FADD23	PE_EMIF_FADD23 pull enable control	R/W	0x1
23:21	D_EMIF_FADD24	000: FADD_24 001: 010: 011: EXT_DMA_REQ_1 100: MPU_UART_RTS2 101: 110: GPIO_76	R/W	0x0
20	PE_EMIF_FADD24	PE_EMIF_FADD24 pull enable control	R/W	0x1
19:17	D_EMIF_FADD25	000: FADD_25 001: clk_13m_req 010: NFCS_0 011: 100: EXT_DMA_REQ_3 101: 110: GPIO_75	R/W	0x0
16	PE_EMIF_FADD25	PE_EMIF_FADD25 pull enable control	R/W	0x1
15:13	D_DDR	000: DQSH, DQSL, SDCLKX 001: , , 010: , , 011: , , 100: , , 101: , , 110: , ,	R/W	0x6
12	PE_DDR	PE_DDR pull enable control	R/W	0x1

Table 5−27. OMAP850 Shared I/O Configuration 5 Register (PERSEUS2_IO_CONF5) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x84				
Bit	**Name**	**Function**	**R/W**	**Reset**
11:9	D_SDCS	000: nSDCS 001: 010: 011: 100: 101: 110: GPIO_71	R/W	0x0
8	PE_SDCS	PE_SDCS pull enable control	R/W	0x1
7:5	D_I2C_SCK	000: MPU_I2C_SCK 001: EXT_IO_3 010: GSM_I2C_SCK 011: 100: 101: USB_SPEED 110: GPIO_70	R/W	0x6
4	PE_I2C_SCK	PE_I2C_SCK pull enable control	R/W	0x1
3:1	D_I2C_SDA	000: MPU_I2C_SDA 001 : EXT_IO_2 010: GSM_I2C_SDA 011: 100: 101: USB_SUSPEND 110: GPIO_69	R/W	0x6
0	PE_I2C_SDA	PE_I2C_SDA pull enable control	R/W	0x1

Table 5−28. OMAP850 Shared I/O Configuration 6 Register (PERSEUS2_IO_CONF6)

Base Address = 0xFFFE:1000, Offset = 0x88				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_NFC_CE2	000: FADD_10 001: CE_2 010: 011: 100: 101: 110: GPIO_90 If PERSEUS25_MODE[3] = 1, then see PERSEUS25_FADD_IOCONF1/2 registers.	R/W	0x0
28	PE_NFC_CE2	PE_NFC_CE2 pull enable control	R/W	0x1

Table 5−28. OMAP850 Shared I/O Configuration 6 Register (PERSEUS2_IO_CONF6) (Continued)

Bit	Name	Function	R/W	Reset
27:25	D_NFC	000: FADD_15, FADD_14, FADD_13, FADD_12, FADD_11, FADD_9, FADD_8, FADD_7, FADD_6, FADD_5, FADD_4, FADD_3, FADD_2, FADD_1 001: ALE, CLE, WE, I/O_7, CE_1, I/O_0, WP, I/O_6, I/O_5, I/O_4, I/O_3, I/O_2, I/O_1, RE 010: , , , , , , , , , , , , , , , , 011: , EXT_DMA_REQ_2, , , , , , , , , , , , 100: MPU_MCSI_CLK, MPU_MCSI_TXD, MPU_MCSI_RXD, MPU_MCSI_FSYNCH, , , , , , , , , , 101: , , , , , , , , , , , , , , , , 110: GPIO_85, GPIO_86, GPIO_87, GPIO_88, GPIO_89, GPIO_91, GPIO_92, GPIO_93, GPIO_94, GPIO_95, GPIO_96, GPIO_97, GPIO_98, GPIO_99 If PERSEUS25_MODE[3] = 1, then see PERSEUS25_FADD_IOCONF1/2 registers.	R/W	0x0
24	PE_NFC	PE_NFC pull enable control	R/W	0x1
23:21	D_EMIF_FADD16	000: FADD_16 001: RDY 010: 011: IO_GSM_14 100: MPU_SPI2_SEN1 101: 110: GPIO_84	R/W	0x0
20	PE_EMIF_FADD16	PE_EMIF_FADD16 pull enable control	R/W	0x1
19:17	D_EMIF_FADD17	000: FADD_17 001: 010: 011: 100: MPU_SPI2_SEN0 101: 110: GPIO_83	R/W	0x0
16	PE_EMIF_FADD17	PE_EMIF_FADD17 pull enable control	R/W	0x1
15:13	D_EMIF_FADD18	000: FADD_18 001: 010: 011: 100: MPU_SPI2_SDI 101: 110: GPIO_82	R/W	0x0
12	PE_EMIF_FADD18	PE_EMIF_FADD18 pull enable control	R/W	0x1

Table 5–28. OMAP850 Shared I/O Configuration 6 Register (PERSEUS2_IO_CONF6) (Continued)

Bit	Name	Function	R/W	Reset
11:9	D_EMIF_FADD19	000: FADD_19 001: 010: 011: 100: MPU_SPI2_SDO 101: 110: GPIO_81	R/W	0x0
8	PE_EMIF_FADD19	PE_EMIF_FADD19 pull enable control	R/W	0x1
7:5	D_EMIF_FADD20	000: FADD_20 001: 010: 011: 100: MPU_SPI2_SCLK 101: 110: GPIO_80	R/W	0x0
4	PE_EMIF_FADD20	PE_EMIF_FADD20 pull enable control	R/W	0x1
3:1	D_EMIF_FADD21	000: FADD_21 001: 010: 011: MPU_UART_TX_IR2 100: MPU_UART_RX2 101: 110: GPIO_79	R/W	0x0
0	PE_EMIF_FADD21	PE_EMIF_FADD21 pull enable control	R/W	0x1

Table 5–29. OMAP850 Shared I/O Configuration7 Register (PERSEUS2_IO_CONF7)

Base Address = 0xFFFE:1000, Offset = 0x8C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_EMIF_FCLK	000: FCLK 001: 010: 011: 100: 101: 110: GPIO_124	R/W	0x0
28	PE_EMIF_FCLK	PE_EMIF_FCLK pull enable control	R/W	0x1
27:25	D_EMIF_RDY	000: NFWAIT 001: RDY 010: 011: 100: 101: 110: GPIO_123	R/W	0x0

Table 5–29. OMAP850 Shared I/O Configuration7 Register (PERSEUS2_IO_CONF7) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x8C				
Bit	**Name**	**Function**	**R/W**	**Reset**
24	PE_EMIF_RDY	PE_EMIF_RDY pull enable control	R/W	0x1
23:21	D_EMIF_NFBAA	000: NFBAA 001: 010: 011: 100: 101: 110: GPIO_121	R/W	0x0
20	PE_EMIF_NFBAA	PE_EMIF_NFBAA pull enable control	R/W	0x1
19:17	D_EMIF	000: NFWE, NFOE, NFWP 001: , , 010: , , 011: , , 100: , , 101: , , 110: GPIO_119, GPIO_120, GPIO_122	R/W	0x0
16	PE_EMIF	PE_EMIF pull enable control	R/W	0x1
15:13	D_EMIF_NFC3	000: NFCS_3 001: 010: NFCS3L 011: 100: 101: 110: GPIO_118	R/W	0x0
12	PE_EMIF_NFC3	PE_EMIF_NFC3 pull enable control	R/W	0x1
11:9	D_EMIF_NFC1	000: NFCS_1 001: 010: NFCS3H 011: IO_GSM_12 100: 101: 110: GPIO_117	R/W	0x0
8	PE_EMIF_NFC1	PE_EMIF_NFC1 pull enable control	R/W	0x1
7:5	D_EMIF_NFC2	000: NFCS_2 001: FADD_25 010: 011: IO_GSM_13 100: 101: 110: GPIO_116	R/W	0x0
4	PE_EMIF_NFC2	PE_EMIF_NFC2 pull enable control	R/W	0x1

Table 5–29. OMAP850 Shared I/O Configuration7 Register (PERSEUS2_IO_CONF7) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x8C				
Bit	**Name**	**Function**	**R/W**	**Reset**
3:1	D_EMIF_NFDATA15_0	000 : FDATA_15, FDATA_14, FDATA_13, FDATA_12, FDATA_11, FDATA_10, FDATA_9, FDATA_8, FDATA_7, FDATA_6, FDATA_5, FDATA_4, FDATA_3, FDATA_2, FDATA_1, FDATA_0 001: , , , , , , , , , , , , , , , , 010: , , , , , , , , , , , , , , , , 011: , , , , , , , , , , , , , , , , 100: , , , , , , , , , , , , , , , , 101: , , , , , , , , , , , , , , , , 110: GPIO_100, GPIO_101, GPIO_102, GPIO_103, GPIO_104, GPIO_105, GPIO_106, GPIO_107, GPIO_108, GPIO_109, GPIO_110, GPIO_111, GPIO_112, GPIO_113, GPIO_114, GPIO_115	R/W	0x0
0	PE_EMIF_NFDATA15_0	PE_EMIF_NFDATA15_0 pull enable control	R/W	0x1

Table 5–30. OMAP850 Shared I/O Configuration 8 Register (PERSEUS2_IO_CONF8)

Base Address = 0xFFFE:1000, Offset = 0x90				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_SPI1_SEN0	000: MPU_SPI1_SEN0 001: FSYNC 010: MPU_MCSI_FSYNCH 011: GSM_UW_nSCS1 100: FSRX1 101: GSM_MCSI_FSYNCH 110: GPIO_132	R/W	0x6
28	PE_SPI1_SEN0	PE_SPI1_SEN0 pull enable control	R/W	0x1
27:25	D_SPI1_SDI	000: MPU_SPI1_SDI 001: SDI 010: MPU_MCSI_RXD 011: GSM_UW_SDI 100: DR1 101: GSM_MCSI_RXD 110: GPIO_131	R/W	0x6
24	PE_SPI1_SDI	PE_SPI1_SDI pull enable control	R/W	0x1
23:21	D_SPI1_SDO	000: MPU_SPI1_SDO 001: SDO 010: MPU_MCSI_TXD 011: GSM_UW_SDO 100: DX1 101: GSM_MCSI_TXD 110: GPIO_130	R/W	0x6
20	PE_SPI1_SDO	PE_SPI1_SDO pull enable control	R/W	0x1

Table 5–30. OMAP850 Shared I/O Configuration 8 Register (PERSEUS2_IO_CONF8) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x90				
Bit	**Name**	**Function**	**R/W**	**Reset**
19:17	D_SPI1_SCLK	000: MPU_SPI1_SCLK 001: SCLK 010: MPU_MCSI_CLK 011: GSM_UW_SCLK 100: CLKRX1 101: GSM_MCSI_CLK 110: GPIO_129	R/W	0x6
16	PE_SPI1_SCLK	PE_SPI1_SCLK pull enable control	R/W	0x1
15:13	D_EMIF_NFRST	000: NFRST 001: 010: 011: 100: 101: 110: GPIO_128	R/W	0x0
12	PE_EMIF_NFRST	PE_EMIF_NFRST pull enable control	R/W	0x1
11:9	D_EMIF_NFBE0	000: NFBE_0 001: 010: BCLKR 011: 100: 101: 110: GPIO_127	R/W	0x0
8	PE_EMIF_NFBE0	PE_EMIF_NFBE0 pull enable control	R/W	0x1
7:5	D_EMIF_NFBE1	000: NFBE_1 001: 010: BCLKX 011: NFCS_0 100: 101: 110: GPIO_126	R/W	0x0
4	PE_EMIF_NFBE1	PE_EMIF_NFBE1 pull enable control	R/W	0x1
3:1	D_EMIF_NFADV	000: NFADV 001: 010: NFCS_0 011: 100: 101: 110: GPIO_125	R/W	0x0
0	PE_EMIF_NFADV	PE_EMIF_NFADV pull enable control	R/W	0x1

Table 5–31. OMAP850 Shared I/O Configuration 9 Register (PERSEUS2_IO_CONF9)

Base Address = 0xFFFE:1000, Offset = 0x94				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_MPU_nIRQ	000: MPU_EXT_nIRQ 001: USB_VBUSI 010: 011: 100: 101: 110: GPIN_4	R/W	0x6
28	PE_MPU_nIRQ	PE_PU_nIRQ pull enable control	R/W	0x1
27:25	D_SMC_PWR	000: SMC_PWCTRL 001: XF 010: TSPACT_9 011: KBC_6 100: GSM_UW_nSCS2 101: MPU_UW_nSCS2 110: GPIO_139 111: HRTFT_CLS	R/W	0x6
24	PE_SMC_PWR	PE_SMC_PWR pull enable control	R/W	0x1
23:21	D_SMC_CD	000: SMC_CD 001: IT_FRAME 010: TSPACT_8 011: TSPACT_11 100: GSM_UW_nSCS1 101: MPU_UW_nSCS1 110: GPIO_138 111: HRTFT_REV	R/W	0x6
20	PE_SMC_CD	PE_SMC_CD pull enable control	R/W	0x1
19:17	D_SMC_RST	000: SMC_RST 001: GSM_IOx 010: TSPACT_7 011: KBC_5 100: GSM_UW_SDO 101: MPU_UW_SDO 110: GPIO_137 111: HRTFT_SPL	R/W	0x6
16	PE_SMC_RST	PE_SMC_RST pull enable control	R/W	0x1
15:13	D_SMC_CLK	000: SMC_CLK 001: EXT_DSP_nIRQ 010: TSPACT_6 011: KBR_6 100: GSM_UW_SCLK 101: MPU_UW_SCLK 110: GPIO_136 111: HRTFT_ASC	R/W	0x6

Table 5–31. OMAP850 Shared I/O Configuration 9 Register (PERSEUS2_IO_CONF9) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x94				
Bit	**Name**	**Function**	**R/W**	**Reset**
12	PE_SMC_CLK	PE_SMC_CLK pull enable control	R/W	0x1
11:9	D_SMC_IO	000: SMC_IO 001: EXT_ARM_nIRQ 010: TSPACT_5 011: KBR_5 100: GSM_UW_SDI 101: MPU_UW_SDI 110: GPIO_135 111: HRTFT_PS	R/W	0x6
8	PE_SMC_IO	PE_SMC_IO pull enable control	R/W	0x1
7:5	D_SPI1_SEN2	000: MPU_SPI1_SEN2 001: CLK13M_IN 010: CRESET 011: MPU_UART_DTR1 100: 101: IO_GSM_3 110: GPIO_134	R/W	0x6
4	PE_SPI1_SEN2	PE_SPI1_SEN2 pull enable control	R/W	0x1
3:1	D_SPI1_SEN1	000: MPU_SPI1_SEN1 001: SEN1 010: 011: GSM_UW_nSCS2 100: CLKS1 101: IO_GSM_2 110: GPIO_133	R/W	0x6
0	PE_SPI1_SEN1	PE_SPI1_SEN1 pull enable control	R/W	0x1

Table 5–32. OMAP850 Shared I/O Configuration 10 Register (PERSEUS2_IO_CONF10)

Base Address = 0xFFFE:1000, Offset = 0x98				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_CLK13MREQ	000: CLK_13M_REQ 001: IO_GSM_3 010: 011: PIPESTAT_3 100: 101: 110: GPIO_145	R/W	0x6
28	PE_CLK13MREQ	PE_CLK13MREQ pull enable control	R/W	0x1

Table 5–32. OMAP850 Shared I/O Configuration 10 Register (PERSEUS2_IO_CONF10) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x98				
Bit	**Name**	**Function**	**R/W**	**Reset**
27:25	D_CLK32K	000: CLK32K 001: 010: 011: 100: 101: 110:	R/W	0x0
24	PE_CLK32K	PE_CLK32k pull enable control	R/W	0x1
23:21	D_TEST_MODE	000: TEST_MODE 001: 010: 011: 100: 101: 110: GPIN_144	R/W	0x0
20	PE_TEST_MODE	PE_TEST_MODE pull enable control	R/W	0x1
19:17	D_NEMU1	000: NEMU1 001: PWL 010: HDQ1W 011: PWT 100: 101: 110: GPIO_143	R/W	0x0
16	PE_NEMU1	PE_NEMU1 pull enable control	R/W	0x1
15:13	D_NEMU0	000: NEMU0 001: Low_power 010: CLK48M_IN 011: TSPEN_3 100: 101: 110: GPIO_142	R/W	0x0
12	PE_NEMU0	PE_NEMU0 pull enable control	R/W	0x1
11:9	D_MPU_LPG2	000: ARM_boot_MLPG2 001: GSM_LPG2 010: RTCK 011: EXT_IO_1 100: 101: 110: GPIO_141	R/W	0x2
8	PE_MPU_LPG2	PE_MPU_LPG2 pull enable control	R/W	0x1

Table 5−32. OMAP850 Shared I/O Configuration 10 Register (PERSEUS2_IO_CONF10) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x98				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:5	D_MPU_LPG1	000: MUX_MODE_MLPG1 001: GSM_LPG1 010: 011: EXT_IO_0 100: NFIQ_PWRFAIL 101: 110: GPIO_140	R/W	0x0
4	PE_MPU_LPG1	PE_MPU_LPG1 pull enable control	R/W	0x1
3:1	D_GSM_nIRQ	000: GSM_EXT_nIRQ 001: 010: 011: 100: 101: 110: GPIN_5	R/W	0x6
0	PE_GSM_nIRQ	PE_GSM_nIRQ pull enable control	R/W	0x1

Table 5−33. OMAP850 Shared I/O Configuration 11 Register (PERSEUS2_IO_CONF11)

Base Address = 0xFFFE:1000, Offset = 0x9C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_CAM_DAT2	000: CAM_DATA_2 001: FSRX1 010: IO_GSM_1 011: TRACEPKT_2 100: GSM_UW_nSCS1 101: DEBUG_5 110: GPIO_153	R/W	0x6
28	PE_CAM_DAT2	PE_CAM_DAT2 pull enable control	R/W	0x1
27:25	D_CAM_DAT1	000: CAM_DATA_1 001: DR1 010: IO_GSM_0 011: TRACEPKT_1 100: GSM_UW_SDO 101: DEBUG_6 110: GPIO_152	R/W	0x6
24	PE_CAM_DAT1	PE_CAM_DAT1 pull enable control	R/W	0x1

Table 5–33. OMAP850 Shared I/O Configuration 11 Register (PERSEUS2_IO_CONF11) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x9C				
Bit	**Name**	**Function**	**R/W**	**Reset**
23:21	D_CAM_DAT0	000: CAM_DATA_0 001: DX1 010: KBC_7 011: TRACEPKT_0 100: GSM_UW_SCLK 101: DEBUG_7 110: GPIO_151	R/W	0x6
20	PE_CAM_DAT0	PE_CAM_DAT0 pull enable control	R/W	0x1
19:17	D_CAM_RSTZ	000: CAM_RSTZ 001: CLKRX1 010: KBC_6 011: PIPESTAT_0 100: GSM_UW_SDI 101: DEBUG_8 110 : GPIO_150	R/W	0x6
16	PE_CAM_RSTZ	PE_CAM_RSTZ pull enable control	R/W	0x1
15:13	D_CAM_VS	000: CAM_VS 001: GSM_MCSI_FSYNCH 010: KBC_5 011: PIPESTAT_1 100: GSM_UW_nSCS2 101: DEBUG_9 110: GPIO_149	R/W	0x6
12	PE_CAM_VS	PE_CAM_VS pull enable control	R/W	0x1
11:9	D_CAM_HS	000: CAM_HS 001: GSM_MCSI_RXD 010: KBR_7 011: PIPESTAT_2 100: ARMIO_2 101: DEBUG_10 110: GPIO_148	R/W	0x6
8	PE_CAM_HS	PE_CAM_HS pull enable control	R/W	0x1
7:5	D_CAM_EXCLK	000: CAM_EXCLK 001: GSM_MCSI_TXD 010: KBR_6 011: TRACESYNC 100: ARMIO_1 101: DEBUG_11 110: GPIO_147	R/W	0x6
4	PE_CAM_EXCLK	PE_CAM_EXCLK pull enable control	R/W	0x1

Table 5−33. OMAP850 Shared I/O Configuration 11 Register (PERSEUS2_IO_CONF11) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x9C				
Bit	**Name**	**Function**	**R/W**	**Reset**
3:1	D_CAM_LCLK	000: CAM_LCLK 001: GSM_MCSI_CLK 010: KBR_5 011: TRACECLK 100: ARMIO_0 101: DEBUG_12 110: GPIO_146	R/W	0x6
0	PE_CAM_LCLK	PE_CAM_LCLK pull enable control	R/W	0x1

Table 5−34. OMAP850 Shared I/O Configuration 12 Register (PERSEUS2_IO_CONF12)

Base Address = 0xFFFE:1000, Offset = 0xA0				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:29	D_KB2	000: KBR_2 001: MPU_I2C_SDA 010: ARMIO_4 011: 100: TRACESYNCB 101: 110: GPIO_161	R/W	0x6
28	PE_KB2	PE_KB2 pull enable control	R/W	0x1
27:25	D_KB1	000: KBR_1 001: EXT_IO_1 010: ARMIO_3 011: 100: PIPESTAT_4 101: 110: GPIO_160	R/W	0x6
24	PE_KB1	PE_KB1 pull enable control	R/W	0x1
23:21	D_KB0	000: KBR_0 001: EXT_IO_0 010: 011: GSM_I2C_SCK 100: PIPESTAT_5 101: 110: GPIO_159	R/W	0x6
20	PE_KB0	PE_KB0 pull enable control	R/W	0x1

Table 5–34. OMAP850 Shared I/O Configuration 12 Register (PERSEUS2_IO_CONF12) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xA0				
Bit	**Name**	**Function**	**R/W**	**Reset**
19:17	D_CAM_DAT7	000: CAM_DATA_7 001: MPU_MCSI_FSYNCH 010: VLYNQ_TXD1 011: TRACEPKT_7 100: SEN1 101: DEBUG_0 110: GPIO_158	R/W	0x6
16	PE_CAM_DAT7	PE_CAM_DAT7 pull enable control	R/W	0x1
15:13	D_CAM_DAT6	000: CAM_DATA_6 001: MPU_MCSI_RXD 010: VLYNQ_RXD1 011: TRACEPKT_6 100: FSYNC 101: DEBUG_1 110: GPIO_157	R/W	0x6
12	PE_CAM_DAT6	PE_CAM_DAT6 pull enable control	R/W	0x1
11:9	D_CAM_DAT5	000: CAM_DATA_5 001: MPU_MCSI_TXD 010: VLYNQ_TXD0 011: TRACEPKT_5 100: SDI 101: DEBUG_2 110: GPIO_156	R/W	0x6
8	PE_CAM_DAT5	PE_CAM_DAT5 pull enable control	R/W	0x1
7:5	D_CAM_DAT4	000: CAM_DATA_4 001: MPU_MCSI_CLK 010: VLYNQ_RXD0 011: TRACEPKT_4 100: SDO 101: DEBUG_3 110: GPIO_155	R/W	0x6
4	PE_CAM_DAT4	PE_CAM_DAT4 pull enable control	R/W	0x1
3:1	D_CAM_DAT3	000: CAM_DATA_3 001: CLKS1 010: IO_GSM_2 011: TRACEPKT_3 100: SCLK 101: DEBUG_4 110: GPIO_154	R/W	0x6
0	PE_CAM_DAT3	PE_CAM_DAT3 pull enable control	R/W	0x1

Table 5–35. OMAP850 Shared I/O Configuration 13 Register (PERSEUS2_IO_CONF13)

Base Address = 0xFFFE:1000, Offset = 0xA4				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:28	RESERVED	RESERVED	R/W	0x0
27:25	D_KB9	000: KBC_4 001: GSM_UW_SCLK 010: VLYNQ_TXD1 011: 100: MPU_SPI2_SEN0 101: EXTERN1_GSM 110: GPIO_168	R/W	0x6
24	PE_KB9	PE_KB9 pull enable control	R/W	0x1
23:21	D_KB8	000: KBC_3 001: GSM_UW_SDI 010: VLYNQ_RXD1 011: 100: MPU_SPI2_SDI 101: EXTERN0_GSM 110: GPIO_167	R/W	0x6
20	PE_KB8	PE_KB8 pull enable control	R/W	0x1
19:17	D_KB7	000: KBC_2 001: GSM_UW_SDO 010: 011: 100: MPU_SPI2_SDO 101: 110: GPIO_166	R/W	0x6
16	PE_KB7	PE_KB7 pull enable control	R/W	0x1
15:13	D_KB6	000: KBC_1 001: EXT_IO_3 010: 011: 100: MPU_SPI2_SEN2 101: 110: GPIO_165	R/W	0x6
12	PE_KB6	PE_KB6 pull enable control	R/W	0x1
11:9	D_KB5	000: KBC_0 001: EXT_IO_2 010: 011: GSM_I2C_SDA 100: MPU_SPI2_SEN1 101: 110: GPIO_164	R/W	0x6
8	PE_KB5	PE_KB5 pull enable control	R/W	0x1

Table 5–35. OMAP850 Shared I/O Configuration 13 Register (PERSEUS2_IO_CONF13) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xA4				
Bit	**Name**	**Function**	**R/W**	**Reset**
7:5	D_KB4	000: KBR_4 001: GSM_UW_nSCS1 010: VLYNQ_RXD0 011: TSPDI 100: MPU_SPI2_SCLK 101: EXTERN1_MPU 110: GPIO_163	R/W	0x6
4	PE_KB4	PE_KB4 pull enable control	R/W	0x1
3:1	D_KB3	000: KBR_3 001: MPU_I2C_SCK 010: VLYNQ_TXD0 011: TSPEN_3 100: 101: EXTERN0_MPU 110: GPIO_162	R/W	0x6
0	PE_KB3	PE_KB3 pull enable control	R/W	0x1

Table 5–36. 48-MHz Input Control Register (PERSEUS_PCC_CONF_REG)

Base Address = 0xFFFE:1000, Offset = 0xB4				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:10	RESERVED	RESERVED	R	0x0
9	PLL_DIV_SEL	Allow to select the divider ratio for the APLL or DPLL clock output. 0: Register in PCC. 1: Register in Tap controller	R/W	0x0
8	PMT_MPU_SEL	Allow to select either OMAP PLL output and Lock or PCC APLL 48 MHz clock and lock for observanility. 0: APLL 48 MHz selected 1: OMAP PLL selected	R/W	0x0
7	PCC_CAM_CLK_REQ	Allow to request the 48 MHz clock the camera interface. 1: request active 0: request inactive	R/W	0x0
6	MCBSP1_CLK_REQ	48 MHz or 13 MHz clock request for MCBSP1. The clock frequency have to be programmed in the PCC. 0: Clock request inactive 1: clock request active	R/W	0x0

Table 5–36. 48-MHz Input Control Register (PERSEUS_PCC_CONF_REG) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xB4				
Bit	**Name**	**Function**	**R/W**	**Reset**
5	MCBSP2_CLK_REQ	48-MHz or 13-MHz clock request for MCBSP2. The clock frequency have to be programmed in the PCC. 0: Clock request inactive. 1: Clock request active.	R/W	0x0
4	UART3_DPLL_REQ	48-MHz clock request for uart3. 0: Clock request inactive. 1: Clock request active.	R/W	0x0
3	UART1_DPLL_REQ	48-MHz clock request for uart1. 0: Clock request inactive. 1: Clock request active.	R/W	0x0
2	PCONF_MMC_DPLL_REQ	48-MHz clock request for MMC/SDIO in MMC mode. 0: Clock request inactive. 1: Clock request active.	R/W	0x0
1	PLL_NCLKEXT_SEL	Allow to select the APLL48MHz and an external 48-MHz clock. 0: External clock 48 MHz. 1: Internal APLL48 MHz.	R/W	0x1
0	EXT13M_CLK_NAPLL13	Allow to select either the APLL 13 MHz output clock either the ext13m_clk(backup). 0: APLL 13 MHz used. 1: ext13m_clk used	R/W	0x0

Table 5–37. BIST Fail Go Register (BIST_STATUS_INTERNAL)

Base Address = 0xFFFE:1000, Offset = 0xB8				
Bit	**Name**	**Function**	**R/W**	**Reset**
31	DONE_COMBINED_INTERNAL	Bist status DONE bit. At 1 when BIST sequence is done.	R	0x0
30	GLOBAL_FAIL_GO_COMBINED_ INTERNAL	Bist status fail bit that combines all fail_go of PERSEUS2 Bists	R	0x1
29:23	UNUSED	Unused	R	0x0
22	TPU_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
21	GEA_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
20	INTBOOTRAM_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
19	SECROM_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
18	SECRAM_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
17	OMAP_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1

Note: All bits are undefined (U) when BIST mode is not activated.

Table 5–37. BIST Fail Go Register (BIST_STATUS_INTERNAL) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xB8				
Bit	**Name**	**Function**	**R/W**	**Reset**
16	TCIF_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
15	ICR_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
14	EAC2_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
13	EAC1_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
12	MMC_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
11	FRAME_BUFFER_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
10	ARM7_CTRL_2_FAIL_GO_INTERNAL	Bist status fail bit (0.5Mb)	R	0x1
9	ARM7_CTRL_1_FAIL_GO_INTERNAL	Bist status fail bit (2Mb)	R	0x1
8:7	USB_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
6:4	UART_MOD_IRDA_FAIL_GO_INTERNAL	Bist status fail bit	R	0x1
3:1	UART_MOD_FAIL_GO_INTERNAL	BIST status fail bit	R	0x1
0	CAMERA_FAIL_GO_INTERNAL	Bist Fail status bit	R	0x1

Note: All bits are undefined (U) when BIST mode is not activated.

Table 5–38. BIST Settings Control Register (BIST_CONTROL)

Base Address = 0xFFFE:1000, Offset = 0xC0				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:30	UNUSED	Unused	R/W	0x0
29	BIST_MODE	BIST mode	R/W	0x0
28	MBIST_TCIF_CTRL_EN	BIST enable signal	R/W	0x0
27	MBIST_ICR_CTRL_EN	BIST enable signal	R/W	0x0
26	MBIST_EAC_CTRL_2_EN	BIST enable signal	R/W	0x0
25	MBIST_EAC_CTRL_1_EN	BIST enable signal	R/W	0x0
24	MBIST_MMC_CTRL_EN	BIST enable signal	R/W	0x0
23	MBIST_USB_HM_CTRL_EN	BIST enable signal	R/W	0x0
22	MBIST_USB_DM_CTRL_EN	BIST enable signal	R/W	0x0
21	MBIST_ARM7_CTRL_2_EN	BIST enable signal	R/W	0x0
20	MBIST_ARM7_CTRL_1_EN	BIST enable signal	R/W	0x0
19	MBIST_INTBOOTRAM_CTRL_EN	BIST enable signal	R/W	0x0
18	MBIST_GEA_CTRL_EN	BIST enable signal	R/W	0x0
17	MBIST_TPU_CTRL_EN	BIST enable signal	R/W	0x0

Table 5−38. BIST Settings Control Register (BIST_CONTROL) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xC0				
Bit	**Name**	**Function**	**R/W**	**Reset**
16:14	MBIST_UART_MOD_IRDA_CTRL_EN	BIST enable signal	R/W	0x0
13	MBIST_SECROM_CTRL_EN	BIST enable signal	R/W	0x0
12	MBIST_SECRAM_CTRL_EN	BIST enable signal	R/W	0x0
11	MBIST_FRAMBUF_CTRL_EN	BIST enable signal	R/W	0x0
10:8	MBIST_UART_MOD_CTRL_EN	BIST enable signal	R/W	0x0
7	MBIST_CAMERA_IF_CTRL_EN	BIST enable signal	R/W	0x0
6	RATIO_EN	0: Ratio value comes from jtag or static value from input pad 1: Ratio value comes from conf	R/W	0x0
5:4	RATIO_DOMAIN_MVE_2MB_RAM	BIST clock divider for GSM internal RAMs (2Mb + 0.5Mb). 00 div by 1 01 div by 2 10 div by 4 11 div by 8	R/W	0x00
3:2	RATIO_DOMAIN_MPU_GSM	BIST clock divider for MPU peripherals and GSM. 00 div by 1 01 div by 2 10 div by 4 11 div by 8	R/W	0x00
1:0	RATIO_DOMAIN_OMAP	BIST clock divider for OMAP and its related peripherals. 00 div by 1 01 div by 2 10 div by 4 11 div by 8	R/W	0x00

Table 5−39. Boot Procedure Register (BOOT_ROM_REG)

Base Address = 0xFFFE:1000, Offset = 0xC4				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	GP_JTAG_REG	Status field for ROM and secure RAM tests	W	0x0000
15:12	UNUSED	Unused	R	0x0
11:9	TEST_SELECTION_INTERNAL	TEST SELECTION: 0x0: Bypass branch (always with DPLL OFF)· 0x1: BURNIN code (always with DPLL OFF)· 0x2: TEST1 0x3: TEST2	R	0x0

Table 5–39. Boot Procedure Register (BOOT_ROM_REG) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xC4				
Bit	**Name**	**Function**	**R/W**	**Reset**
8	DPLL_CONFIG_INTERNAL	DPLL configuration· 0x0: DPLL OFF (DPLL bypassed)· 0x1: DPLL ON. (x 10 –> for ROMed tests only)	R	0x0
7:0	OMAP850_TI_TEST_SEL_ INTERNAL	ROMed test cases, LED/EMIFS and selection: 0x00: OMAP850 BOOT ROM 0xA5: TI_TEST function selection	R	0x00

Table 5–40. Secure chip (PRODUCTION_ID_REG)

Base Address = 0xFFFE:1000, Offset = 0xC8				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:30	GP_DEVICE	ES1.0: NA ES1.x: Select the general-purpose or the normal DEVICE boot 00: GP_DEVICE boot 11: Normal boot (high-security device)	R	0x0
29	R_PROT	0: Protect DSP ROM 1: Unprotect DSP ROM ES1.0: NA	R	0x0
28:26	UNUSED	Unused	R	0x0
25	VBOX_EN	0: DFT read, DFT write values come from EFUSE. 1: DFT read, DFT write values are set by software.	R/W	0x0
24:9	FUSE_COMPARE_REG_ INTERNAL	This register is always set to 0x5555 by efuse. If this value is identical to the reference value, the INITZ sequence is OK. If not, a signal blocks all the chip by setting the ULPD out reset to 0	R	0x5555
8	PROTECT_ CTLSECUREDATA_INTERNAL	Bit that protect the EFUSEs for any reprogrammation. –> GATING of EFUSE datain and dataout	R	0x0
7	DFT_WRITE_OMAP	Default DFT_WRITE OMAP value	R/W	0x0
6	DFT_READ_OMAP	Default OMAP DFT_READ value	R/W	0x0
5	DFT_WRTE_MGS3	Not used on OMAP850	R	0x0

Table 5–40. Secure chip (PRODUCTION_ID_REG) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xC8				
Bit	**Name**	**Function**	**R/W**	**Reset**
4	DFT_READ_MGS3	Not used on OMAP850	R	0x0
3:0	TST_DEVICE_TYPE	0000 not programmed 0001 normal 0010 normal 0011 normal 0100 emulator 0101 bad 0110 bad 0111 bad 1000 emulator 1001 bad 1010 bad 1011 bad 1100 emulator 1101 bad 1110 bad 1111 bad	R	0x0

Table 5–41. Secure ROM Signature 1 Register (BIST_SECROM_SIGNATURE1_INTERNAL)

Base Address = 0xFFFE:1000, Offset = 0xD0				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	BIST_SECROM_ SIGNATURE1_INTERNAL	Secure ROM signature in BIST mode	R	0x0

Table 5–42. Secure ROM Signature 2 Register (BIST_SECROM_SIGNATURE2_INTERNAL)

Base Address = 0xFFFE:1000, Offset = 0xD4				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:0	BIST_SECROM_ SIGNATURE2_INTERNAL	Secure ROM signature in BIST mode	R	0X0

Table 5–43. BIST Settings Control Register (BIST_CONTROL_2)

Base Address = 0xFFFE:1000, Offset = 0xD8				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:9	UNUSED	UNUSED	R/W	0x00000
8	MBIST_HOLD	signal used for scan purpose	R/W	0x0
7:6	MBIST_SETUP	signals used for IDDQ purpose	R/W	0x00
5:4	MBIST_ALGO_MODE	Signals used for IDDQ purpose	R/W	0x00
3	MBIST_TCK_MODE	Use TCK clock for data logging purpose. Active at 1.	R/W	0x0

Table 5–43. BIST Settings Control Register (BIST_CONTROL_2) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xD8				
Bit	**Name**	**Function**	**R/W**	**Reset**
2	MBIST_DIAG_EN	Mux COMPSTAT functionnality on BIST_GO signals This feature depends on the type of BIST controller. Active at 1	R/W	0x0
1	MBIST_DL_EN	Enable data logging logic Active at 1	R/W	0x1
0	MBIST_RST_MEM	Filler 0 of the BIST controller Active at 1	R/W	0x0

Table 5–44.Debug Signal Selection 1 Register (DEBUG1)

Base Address = 0xFFFE:1000, Offset = 0xE0				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:28	OBS_MUX7	Selection of debug signals: 0000: mode0 1011: Mode11	R/W	0x0
27:24	OBS_MUX6	Selection of debug signals: 0000: Mode0 1011: Mode11	R/W	0x0
23:20	OBS_MUX5	Selection of debug signals: 0000: Mode0 1011: Mode11	R/W	0x0
19:16	OBS_MUX4	Selection of debug signals: 0000: Mode0 1011: Mode11	R/W	0x0
15:12	OBS_MUX3	Selection of debug signals: 0000: Mode0 1011: Mode11	R/W	0x0
11:8	OBS_MUX2	Selection of debug signals: 0000: Mode0 1011: Mode11	R/W	0x0
7:4	OBS_MUX1	Selection of debug signals: 0000: Mode0 1011: Mode11	R/W	0x0
3:0	OBS_MUX0	Selection of debug signals: 0000: Mode0 1011: Mode11	R/W	0x0

Table 5–45. Debug Signal Selection 2 Register (DEBUG2)

Base Address = 0xFFFE:1000, Offset = 0xE4				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:20	RESERVED	RESERVED	R/W	0x0
19:16	OBS_MUX12	Selection of debug signals: 0000: Mode0 1011: Mode11	R/W	0x0
15:12	OBS_MUX11	Selection of debug signals: 0000: Mode0 1011: Mode11	R/W	0x0
11:8	OBS_MUX10	Selection of debug signals: 0000: Mode0 1011: Mode11	R/W	0x0
7:4	OBS_MUX9	Selection of debug signals: 0000: Mode0 1011: Mode11	R/W	0x0
3:0	OBS_MUX8	Selection of debug signals: 0000: Mode0 1011: Mode11	R/W	0x0

Table 5–46.DMA and IRQ Selection Register (DEBUG_DMA_IRQ)

Base Address = 0xFFFE:1000, Offset = 0xE8				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:22	RESERVED	RESERVED	R/W	0X0
21:16	OBS_IRQ2_SEL	Selection of 1 IRQ (level 2) out of 64 for observability: 000000: IRQ 0 selected 111111: IRQ 63 selected	R/W	0x0
15	RESERVED	RESERVED	R/W	0x0
14:10	OBS_IRQ1_SEL	Selection of 1 IRQ (level 1) out of 32 for observability: 00000: IRQ 0 selected 11111: IRQ 31 selected	R/W	0x0
9:8	RESERVED	RESERVED	R/W	0x0
7:4	OBS_DMA_REQ_TX_SEL	Selection of 1 DMA TX request out of 16 for observability: 0000: DMA TX request 0 selected 1111: DMA TX request 15 selected	R/W	0x0
3:0	OBS_DMA_REQ_RX_SEL	Selection of 1 DMA RX request out of 16 for observability: 0000: DMA RX request 0 selected 1111: DMA RX request 15 selected	R/W	0x0

Table 5–47. Software Compatibility Purpose Register (PERSEUS25_MODE)

Base Address = 0xFFFE:1000, Offset = 0xEC				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:16	P25_MODE_EXT_ DMAREQ_SEL	Defines if the following DMA requests are internal (corresponding bit set to) or external (corresponding bit set to 1). The number of the external bit request used is defined by pin multiplexing configuration. Bit 16 = 0 : DMA_REQ_2 is connected to MCSI_RX DMA request. Bit 16 = 1 : DMA_REQ_2 is connected to external DMA Request. Bit 17 = 0 : DMA_REQ_3 is connected to I2C_RX DMA request. Bit 17 = 1 : DMA_REQ_3 is connected to external DMA Request. Bit 18 = 0 : DMA_REQ_5 is connected to VLYNQ DMA request. Bit 18 = 1 : DMA_REQ_5 is connected to external DMA Request. Bit 19 = 0 : DMA_REQ_6 is connected to SHA1/MD5 DMA request. Bit 19 = 1 : DMA_REQ_6 is connected to external DMA Request. Bit 20 = 0 : DMA_REQ_7 is connected to UWIRE DMA request. Bit 20 = 1 : DMA_REQ_7 is connected to external DMA Request. Bit 21 = 0 : DMA_REQ_9 is connected to MCSBP1_RX DMA request. Bit 21 = 1 : DMA_REQ_9 is connected to external DMA Request. Bit 22 = 0 : DMA_REQ_10 is connected to MCBSP2_TX DMA request. Bit 22 = 1 : DMA_REQ_10 is connected to external DMA Request. Bit 23 = 0 : DMA_REQ_13 is connected to UART_MODEM1_RX DMA request. Bit 23 = 1 : DMA_REQ_13 is connected to external DMA Request. Bit 24 = 0 : DMA_REQ_14 is connected to UART_MODEM_IRDA2_TX DMA request. Bit 24 = 1 : DMA_REQ_14 is connected to external DMA Request. Bit 25 = 0 : DMA_REQ_16 is connected to DES3DES IN DMA request. Bit 25 = 1 : DMA_REQ_16 is connected to external DMA Request.	R/W	0x0000

Table 5–47. Software Compatibility Purpose Register (PERSEUS25_MODE) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xEC				
Bit	**Name**	**Function**	**R/W**	**Reset**
		Bit 26 = 0 : DMA_REQ_19 is connected to SMC_RX DMA request. Bit 26 = 1 : DMA_REQ_19 is connected to external DMA Request. Bit 27 = 0 : DMA_REQ_20 is connected to CAM-ERA_RX DMA request. Bit 27 = 1 : DMA_REQ_20 is connected to external DMA Request. Bit 28 = 0 : DMA_REQ_21 is connected to MMC_TX DMA request. Bit 28 = 1 : DMA_REQ_21 is connected to external DMA Request. Bit 29 = 0 : DMA_REQ_23 is connected to NDFLASH End_Of_Burst DMA request. Bit 29 = 1 : DMA_REQ_23 is connected to external DMA Request. Bit 30 = 0 : DMA_REQ_28 is connected to USB_OTG_RX2 DMA request. Bit 30 = 1 : DMA_REQ_28 is connected to external DMA Request. Bit 31 = 0 : DMA_REQ_31 is connected to USB_OTG_TX2 DMA request. Bit 31 = 1 : DMA_REQ_31 is connected to external DMA Request.		
15:4	RESERVED	Reserved bits	R	0x0
3	FADD15_1_MAPSEL	FADD[15:1] mapping selection 0: FADD[15:1] multiplexing selected by P2 IO CONF 1: FADD[5:1] individual multiplexing selected by new IO_CONF	R/W	0x0
2	FB_RESP	Framebuffer OCP response latency control 0: Synchronous response (1 cycle latency) 1: Asynchronous response (no latency)	R/W	0x1
1	FB_CLOCK_EN	Framebuffer clock control 0: Framebuffer is gated 1: Framebuffer clock is running	R/W	0x1
0	CAMERA_OCPT_SEL	Control of camera mapping on OCP-T interface 0: Camera is located on OCP-T1 port 1: Camera is located on OCP-T2 port	R/W	0x1

Table 5–48. 8 to 1 Address Configuration Register(PERSEUS25_FADD_IOCONF1)

Base Address = 0xFFFE:1000, Offset = 0xF0				
Bit	**Name**	**Function**	**R/W**	**Reset**
31	FADD_8_PU_EN	FADD_8 enable control	R/W	0x1
30:28	FADD_8	000: FADD_8 001: WP 010: 011: 100: 101: 110: GPIO_92	R/W	0x0
27	FADD_7_PU_EN	FADD_7 enable control	R/W	0x1
26:24	FADD_7	000: FADD_7 001: I/O_6 010: 011: 100: 101: 110: GPIO_93	R/W	0x0
23	FADD_6_PU_EN	FADD_6 enable control	R/W	0x1
22:20	FADD_6	000: FADD_6 001: I/O_5 010: 011: 100: 101: 110: GPIO_94	R/W	0x0
19	FADD_5_PU_EN	FADD_5 enable control	R/W	0x1
18:16	FADD_5	000: FADD_5 001: I/O_4 010: 011: 100: 101: 110: GPIO_95	R/W	0x0
15	FADD_4_PU_EN	FADD_4 enable control	R/W	0x1
14:12	FADD_4	000: FADD_4 001: I/O_3 010: 011: 100: 101: 110: GPIO_96	R/W	0x0
11	FADD_3_PU_EN	FADD_3 enable control	R/W	0x1

Table 5−48. 8 to 1 Address Configuration Register(PERSEUS25_FADD_IOCONF1) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xF0				
Bit	**Name**	**Function**	**R/W**	**Reset**
10:8	FADD_3	000: FADD_3 001: I/O_2 010: 011: 100: 101: 110: GPIO_97	R/W	0x0
7	FADD_2_PU_EN	FADD_2 enable control	R/W	0x1
6:4	FADD_2	000: FADD_2 001: I/O_1 010: 011: 100: 101: 110: GPIO_98	R/W	0x0
3	FADD_1_PU_EN	FADD_1 enable control	R/W	0x1
2:0	FADD_1	000: FADD_1 001: RE 010: 011: 100: 101: 110: GPIO_99	R/W	0x0

Table 5−49. 15 to 9 Address Configuration Register (PERSEUS25_FADD_IOCONF2)

Base Address = 0xFFFE:1000, Offset = 0xF4				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:28	RESERVED	Reserved bits	R/W	0x0
27	FADD_15_PU_EN	FADD_15 enable control	R/W	0x1
26:24	FADD_15	000: FADD_15 001: ALE 010: 011: 100: 101: 110: GPIO_85	R/W	0x0
23	FADD_14_PU_EN	FADD_14 enable control	R/W	0x1

Table 5–49. 15 to 9 Address Configuration Register (PERSEUS25_FADD_IOCONF2) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xF4				
Bit	**Name**	**Function**	**R/W**	**Reset**
22:20	FADD_14	000: FADD_14 001: CLE 010: 011: 100: 101: 110: GPIO_86	R/W	0x0
19	FADD_13_PU_EN	FADD_13 enable control	R/W	0x1
18:16	FADD_13	000: FADD_13 001: WE 010: 011: 100: 101: 110: GPIO_87	R/W	0x0
15	FADD_12_PU_EN	FADD_12 enable control	R/W	0x1
14:12	FADD_12	000: FADD_12 001: I/O_7 010: 011: 100: 101: 110: GPIO_88	R/W	0x0
11	FADD_11_PU_EN	FADD_11 enable control	R/W	0x1
10:8	FADD_11	000: FADD_11 001: CE_1 010: 011: 100: 101: 110: GPIO_89	R/W	0x0
7	FADD_10_PU_EN	FADD_10 enable control	R/W	0x1
6:4	FADD_10	000: FADD_10 001: CE_2 010: 011: 100: 101: 110: GPIO_90	R/W	0x0

Table 5–49. 15 to 9 Address Configuration Register (PERSEUS25_FADD_IOCONF2) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xF4				
Bit	**Name**	**Function**	**R/W**	**Reset**
3	FADD_9_PU_EN	FADD_9 enable control	R/W	0x1
2:0	FADD_9	000: FADD_9 001: I/O_0 010: 011: 100: 101: 110: GPIO_91	R/W	0x0

Table 5–50. TWC310 Control Register (PERSEUS25_DAGON_MODE)

Base Address = 0xFFFE:1000, Offset = 0xF8				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:3	RESERVED	RESERVED bits	R	0x0
2:1	DG_EMU	00: DAGON emulation port on debug pins 01: DAGON mono-emulation port on Perseus2.5 emulation port 10: DAGON emulation linked to Perseus2.5 emulation (Dagon added at end of Perseus2.5 emulation chain 11: Reserved	R/W	0x0
0	INTERNAL_DAGON	0: Connection of GSM-S to external Dagon 1: Connection of GSM-S to internal Dagon	R/W	0x0

Table 5–51. OMAP850 Shared I/O Configuration 0 Register (PERSEUS25_DAGON_IO_CONF0)

Base Address = 0xFFFE:1000, Offset = 0xFC				
Bit	**Name**	**Function**	**R/W**	**Reset**
31	PE_D_DEBUG_0_UB	pe_D_debug_0_ub pull enable control	R/W	0x1
30	PE_D_DEBUG_1_UB	pe_D_debug_1_ub pull enable control	R/W	0x1
29	PE_D_DEBUG_2_UB	pe_D_debug_2_ub pull enable control	R/W	0x1
28	PE_D_TDO_UB	pe_D_tdo_ub pull enable control	R/W	0x1
27	PE_D_NEMU0_UB	pe_D_nemu0_ub pull enable control	R/W	0x1
26	PE_D_NEMU1_UB	pe_D_nemu1_ub pull enable control	R/W	0x1
25	PE_D_BDX_UB	pe_D_bdx_ub pull enable control	R/W	0x1
24	PE_D_BFSX_UB	pe_D_bfsx_ub pull enable control	R/W	0x1
23	PE_D_BDR_UB	pe_D_bdr_ub pull enable control	R/W	0x1
22	PE_D_BFSR_UB	pe_D_bfsr_ub pull enable control	R/W	0x1

Table 5–51. OMAP850 Shared I/O Configuration 0 Register (PERSEUS25_DAGON_IO_CONF0) (Continued)

Base Address = 0xFFFE:1000, Offset = 0xFC				
Bit	**Name**	**Function**	**R/W**	**Reset**
21	PE_D_DEBUG_3_UB	pe_D_debug_3_ub pull enable control	R/W	0x1
20	PE_D_IN_INT_0_UB	pe_D_in_int_0_ub pull enable control	R/W	0x1
19	PE_D_IN_INT_1_UB	pe_D_in_int_1_ub pull enable control	R/W	0x1
18	PE_D_OUT_INT_0_UB	pe_D_out_int_0_ub pull enable control	R/W	0x1
17	PE_D_IN_INT_2_UB	pe_D_in_int_2_ub pe_D_in_int_2_ub	R/W	0x1
16	PE_D_OUT_INT_1_UB	pe_D_out_int_1_ub pull enable control	R/W	0x1
15	PE_D_DEBUG_4_UB	pe_D_debug_4_ub pull enable control	R/W	0x1
14	PE_D_RX_UB	pe_D_rx_ub pull enable control	R/W	0x1
13	PE_D_TX_UB	pe_D_tx_ub pe_D_tx_ub	R/W	0x1
12	PE_D_RTS_UB	pe_D_rts_ub pull enable control	R/W	0x1
11	PE_D_CTS_UB	pe_D_cts_ub pull enable control	R/W	0x1
10	PE_D_DEBUG_5_UB	pe_D_debug_5_ub pull enable control	R/W	0x1
9	PE_D_DEBUG_6_UB	pe_D_debug_6_ub pull enable control	R/W	0x1
8	PE_D_DEBUG_7_UB	pe_D_debug_7_ub pull enable control	R/W	0x1
7	PE_D_SPARE_23_UB	pe_D_spare_23_ub pull enable control	R/W	0x1
6	PE_D_SPARE_24_UB	pe_D_spare_24_ub pull enable control	R/W	0x1
5	PE_D_DEBUG_8_UB	pe_D_debug_8_ub pull enable control	R/W	0x1
4	PE_D_DEBUG_9_UB	pe_D_debug_9_ub pull enable control	R/W	0x1
3	PE_D_SPARE_25_UB	pe_D_spare_25_ub pull enable control	R/W	0x1
2	PE_D_DEBUG_10_UB	pe_D_debug_10_ub pull enable control	R/W	0x1
1	PE_D_DEBUG_11_UB	pe_D_debug_11_ub pull enable control	R/W	0x1
0	PE_D_MCSI_CLK_UB	pe_D_mcsi_clk_ub pull enable control	R/W	0x1

Table 5–52. OMAP850 Shared I/O Configuration 1 Register (PERSEUS2_DAGON_IO_CONF1)

Base Address = 0xFFFE:1000, Offset = 0x100				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:4	UNUSED	Unused bits	R/W	0x1
3	PE_D_CLK13M_REQ_UB	pe_D_clk13m_req_ub pull enable control	R/W	0x1
2	PE_D_MCSI_RXD_UB	pe_D_mcsi_rxd_ub pull enable control	R/W	0x1

Table 5−52. OMAP850 Shared I/O Configuration 1 Register (PERSEUS2_DAGON_IO_CONF1) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x100				
Bit	**Name**	**Function**	**R/W**	**Reset**
1	PE_D_MCSI_FSYNCH_UB	pe_D_mcsi_fsynch_ub pull enable control	R/W	0x1
0	PE_D_MCSI_TXD_UB	pe_D_mcsi_txd_ub pull enable control	R/W	0x1

Table 5−53. TWC310 Dual Debug Signal Selection 0 Register (DEBUG_DAGON_0)

Base Address = 0xFFFE:1000, Offset = 0x104				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:30	OBS_DAGON_MUX15	Selection of debug signals: 01: Mode1 11: Mode3	R/W	0x01
29:28	OBS_DAGON_MUX14	Selection of debug signals: 00: Mode0 01: Mode1 10: Mode2 11: Mode3	R/W	0x01
27:26	OBS_DAGON_MUX13	Selection of debug signals: 11: Mode3	R/W	0x01
25:24	OBS_DAGON_MUX12	Selection of debug signals: 10: Mode2 11: Mode3	R/W	0x01
23:22	OBS_DAGON_MUX11	Selection of debug signals: 00: Mode0 01: Mode1 11: Mode3	R/W	0x01
21:20	OBS_DAGON_MUX10	Selection of debug signals: 10: Mode2 11: Mode3	R/W	0x01
19:18	OBS_DAGON_MUX9	Selection of debug signals: 00: Mode0 01: Mode1 10: Mode2 11: Mode3	R/W	0x01
17:16	OBS_DAGON_MUX8	Selection of debug signals: 00: Mode0 01: Mode1 10: Mode2 11: Mode3	R/W	0x01
15:14	OBS_DAGON_MUX7	Selection of debug signals: 00: Mode0 01: Mode1 10: Mode2	R/W	0x01

Table 5–53. TWC310 Dual Debug Signal Selection 0 Register (DEBUG_DAGON_0) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x104				
Bit	**Name**	**Function**	**R/W**	**Reset**
13:12	OBS_DAGON_MUX6	Selection of debug signals: 00: Mode0 01: Mode1 10: Mode2 11: Mode3	R/W	0x01
11:10	OBS_DAGON_MUX5	Selection of debug signals: 00: Mode0 01: Mode1 10: Mode2	R/W	0x01
9:8	OBS_DAGON_MUX4	Selection of debug signals: 00: Mode0 01: Mode1 10: Mode2	R/W	0x01
7:6	OBS_DAGON_MUX3	Selection of debug signals: 00: Mode0 01: Mode1 10: Mode2	R/W	0x01
5:4	OBS_DAGON_MUX2	Selection of debug signals: 00: Mode0 01: Mode1	R/W	0x01
3:2	OBS_DAGON_MUX1	Selection of debug signals: 00: Mode0 01: Mode1	R/W	0x01
1:0	OBS_DAGON_MUX0	Selection of debug signals: 00: Mode0 01: Mode1	R/W	0x01

Table 5–54. TWC310 Dual Debug Signal Selection 1 Register (DEBUG_DAGON_1)

Base Address = 0xFFFE:1000, Offset = 0x108				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:8	UNUSED	Unused bits	R/W	0x0
7:6	OBS_DAGON_MUX19	Selection of debug signals: 00: Mode0 01: Mode1	R/W	0x01
5:4	OBS_DAGON_MUX18	Selection of debug signals: 01: Mode1	R/W	0x01

Table 5−54. TWC310 Dual Debug Signal Selection 1 Register (DEBUG_DAGON_1) (Continued)

Base Address = 0xFFFE:1000, Offset = 0x108				
Bit	**Name**	**Function**	**R/W**	**Reset**
3:2	OBS_DAGON_MUX17	Selection of debug signals: 00: Mode0 01: Mode1	R/W	0x01
1:0	OBS_DAGON_MUX16	Selection of debug signals: 01: Mode1 11: Mode3	R/W	0x01

Table 5−55. OMAP850 Shared I/O Configuration Register (PCONF_DAGON_JTAG_CTRL)

Base Address = 0xFFFE:1000, Offset = 0x10C				
Bit	**Name**	**Function**	**R/W**	**Reset**
31:1	UNUSED	UNUSED bits	R/W	0x0
0	DAGON_JTAG_SWITCH	0: Functional 1: Observability	R/W	0x0

Chapter 6

MPU-S Memory Mapping

This chapter describes the shared memory and memory mapping for the MPU-Subsystem (MPU-S) of the OMAP850 multimedia processor.

Topic **Page**

6.1 MPU-S Memory Mapping 6-2

6.1 MPU-S Memory Mapping

The external memory space is shared by the ARM926EJS (MPU) and the DMA controller. Shared data can be stored in this data space. Because the cache coherency feature is not supported by the cache controller of ARM926EJS, the shared data space must be used as noncacheable space.

6.1.1 MPU Memory Space

Seven chip-selects are provided for the external memory devices and the internal memory. A 64M-byte address range is available (except for the external SDRAM bus and the local bus) for each chip-select. To minimize the design complexity and optimize memory access time, some chip-selects are dedicated to devices plugged into specific buses. Figure 6–1 shows the MPU memory map and Table 6–1 describes the chip-selects associated with the memory bus.

Figure 6−1. MPU Memory Maps

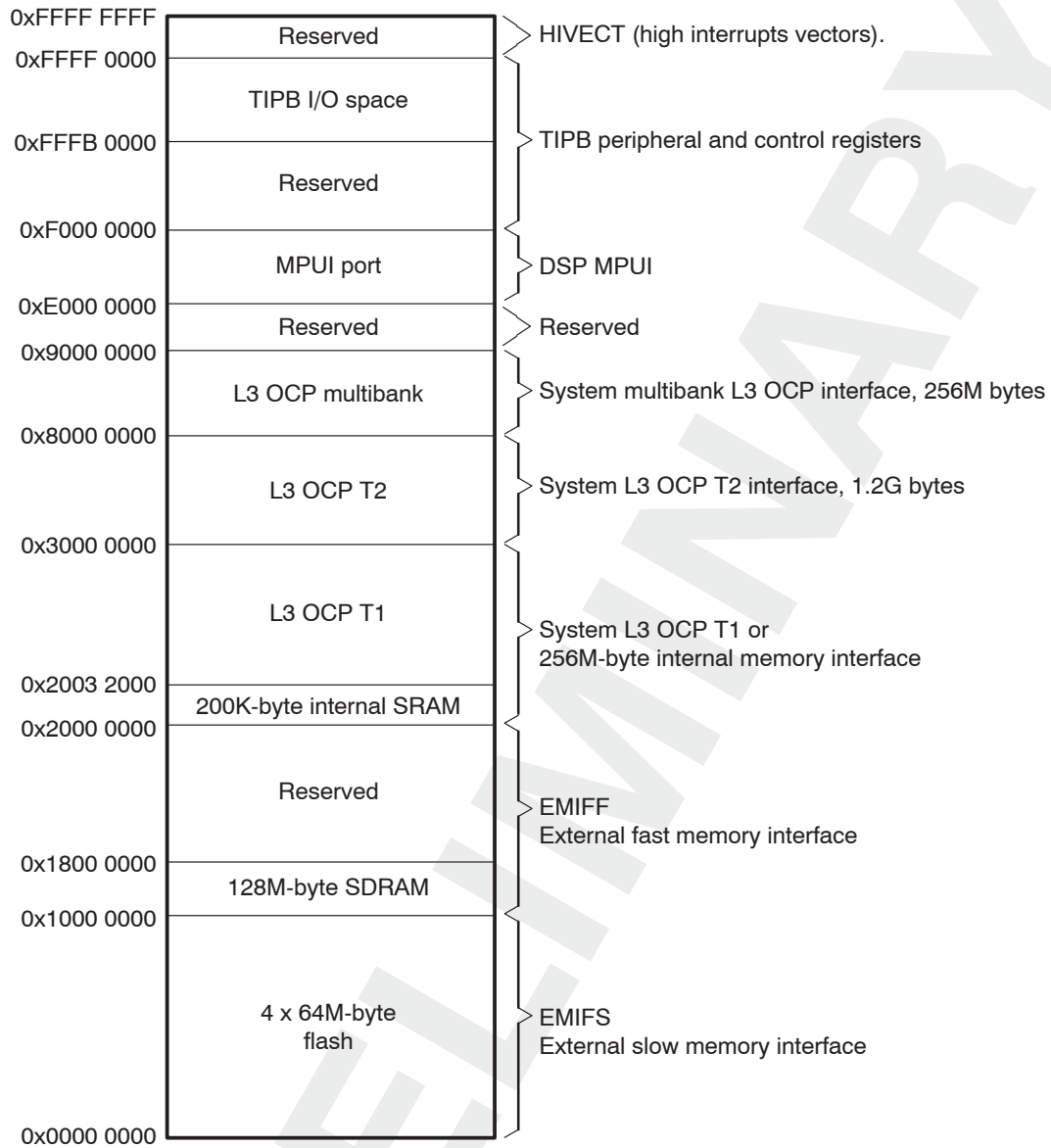


Table 6–1. Memory Bus Associated With Chip-Selects

Device Name		Start Address	Stop Address	Size	Data Access
Flash, SDRAM and OCP Space					
			Slow Memory Interface (EMIFS)		
nCS0	Flash	0000:0000	03FF:FFFF	64M bytes	8/16/32
nCS1	Flash	0400:0000	07FF:FFFF	64M bytes	8/16/32
nCS2	Flash	0800:0000	0BFF:FFFF	64M bytes	8/16/32
nCS3	Flash	0C00:0000	0FFF:FFFF	64M bytes	8/16/32
			Fast Memory Interface (EMIFF)		
nCS4	SDRAM	1000:0000	17FF:FFFF	128M bytes	8/16/32
Reserved		1800:0000	1FFF:FFFF		
			OCP Targets		
OCP-T1		2000:0000	2FFF:FFFF	256M bytes	8/16/32
OCP-T2		3000:0000	7FFF:FFFF	1.2G bytes	8/16/32
OCP-MB		8000:0000	8FFF:FFFF	256M bytes	8/16/32

Internal and external peripherals are mapped to the ARM926EJS memory space in two different sections. These spaces are accessible through STROBE 1 and STROBE 0 with a range of 2K bytes for each external peripheral, allowing connection with up to 122 external devices (two MPU TIPB bridges).

Table 6–2 and Table 6–3 describe the MPU memory space and the TIPB peripherals address space, respectively.

Table 6–2. MPU Memory Address Space

Device Name		Start Address	Stop Address	Size	Data Access
			Slow Memory Interface (EMIFS)		
nCS0		0000:0000	03FF:FFFF	64M bytes	16/32 R/W
	Boot ROM	0000:0000	0000:FFFF	64K bytes	32-bit execute only
	Reserved boot ROM	0001:0000	0003:FFFF	192K bytes	32-bit execute only
	Reserved	0004:0000	001F:FFFF		
	Secure RAM	0020:0000	0020:3FFF	16K bytes	16-bit secure R/W
	Reserved	0020:4000	0020:FFFF		
	Secure eFuse chain1	0021:0000	0021:000F	128 bits	Read only
	Secure eFuse chain2	0021:0010	0021:002F	256 bits	Read only
	Reserved	0021:0030	01FF:FFFF		
	NOR flash	0200:0000	03FF:FFFF	32M bytes	16/32 R/W

Table 6–2. MPU Memory Address Space (Continued)

Device Name		Start Address	Stop Address	Size	Data Access
Slow Memory Interface (EMIFS) (Continued)					
nCS1	NOR flash	0400:0000	07FF:FFFF	64M bytes	16/32 R/W
nCS2	NOR flash	0800:0000	0BFF:FFFF	64M bytes	16/32 R/W
nCS3	NOR flash	0C00:0000	0FFF:FFFF	64M bytes	16/32 R/W
Fast Memory Interface (EMIFF)					
nCS4	SDRAM	1000:0000	17FF:FFFF	128M bytes	16 R/W
Reserved		1800:0000	1FFF:FFFF		
L3 OCP T1					
SRAM		2000:0000	2003:1FFF	200K bytes	32 R/W
Reserved		2003:2000	2003:FFFF		
Camera IF		2004:0000	2004:0FFF	2K bytes	32 R/W
Reserved		2004:1000	2FFF:FFFF		
L3 OCP T2					
Reserved		3000:0000	3000:0FFF		
Reserved		3000:1000	3000:1FFF		
VLYNQ registers		3000:2000	3000:21FF	512 bytes	32 R/W
Reserved		3000:2200	30FF:FFFF		
VLYNQ TX		3100:0000	34FF:FFFF	64M bytes	32 R/W
Reserved		3500:0000	3FFF:FFFF		
Camera IF		4000:0000	4000:0FFF	2K bytes	32 R/W
Reserved		4000:1000	7FFF:FFFF		

Table 6–3. TIPB Peripherals Address Space

Device Name		Start Address	End Address	Size (Bytes)	Data Access	Address Align-ment	Address Compatibility
Reserved		F0000:0000	FFFA:FFFF				
Public TIPB Bridge							
STROBE0, FFFB:0000 -> FFFB:FFFF							
CS = 0		FFFB:0000	FFFB:07FF	2K			
	UART_MODEM 1	FFFB:0000	FFFB:03FF	1K	8	8	P1, H1, H2
	USB OTG	FFFB:0400	FFFB:07FF	1K	32	32	H2

Table 6–3. TIPB Peripherals Address Space (Continued)

Device Name		Start Address	End Address	Size (Bytes)	Data Access	Address Align-ment	Address Compatibility
Public TIPB Bridge (Continued)							
STROBE0, FFFB:0000 -> FFFB:FFFF (Continued)							
CS = 1	UART_MODEM_ IRDA 2	FFFB:0800	FFFB:0FFF	2K	8	8	P1, H1, H2
CS = 2	McBSP1	FFFB:1000	FFFB:17FF	2K	16	16	P1, H1, H2
CS = 3	McBSP2	FFFB:1800	FFFB:1FFF	2K	16	16	H1
CS = 4	MCSI	FFFB:2000	FFFB:27FF	2K	16	16	None
CS = 5	NAND_FLASH	FFFB:2800	FFFB:2FFF	2K	32	32	None
CS = 6	µWire	FFFB:3000	FFFB:37FF	2K	16	16	P1, H1
CS = 7	I^2C	FFFB:3800	FFFB:3FFF	2K	16	16	P1, H1, H2
CS = 8		FFFB:4000	FFFB:47FF	2K			
	USB client	FFFB:4000	FFFB:43FF	1K	32	32	P1, H1, H2
	Reserved	FFFB:4400	FFFB:47FF	1K			
CS = 9	RTC	FFFB:4800	FFFB:4FFF	2K	8	8	P1, H1, H2
CS = 10	MPUIO	FFFB:5000	FFFB:57FF	2K	16	16	H1, H2
CS = 11	PWL	FFFB:5800	FFFB:5FFF	2K	8	8	H1, H2
CS = 12	PWT	FFFB:6000	FFFB:67FF	2K	8	8	H1, H2
CS = 13	Reserved	FFFB:6800	FFFB:6FFF	2K			
CS = 14	HDQ_1WIRE	FFFB:7000	FFFB:77FF	2K	32	32	None
CS = 15	MMC_SDIO	FFFB:7800	FFFB:7FFF	2K	16	16	P1, H1, H2
CS = 16	Reserved	FFFB:8000	FFFB:87FF	2K			
CS = 17	SMC	FFFB:8800	FFFB:8FFF	2K	16	16	None
CS = 18	TIMER32K	FFFB:9000	FFFB:97FF	2K	32	32	P1, H1, H2
CS = 19	DUAL_MODE_ TIMER	FFFB:9800	FFFB:9FFF	2K	32	32	None
CS = 20		FFFB:A000	FFFB:A7FF	2K			
	USB host	FFFB:A000	FFFB:A3FF	1K	32	32	H1, H2
	Reserved	FFFB:4000	FFFB:47FF	1K			
CS = 21	LPG	FFFB:A800	FFFB:AFFF	2K	8	8	P1
CS = 22	EAC	FFFB:B000	FFFB:B7FF	2K	16	16	P1

Table 6–3. TIPB Peripherals Address Space (Continued)

Device Name	Start Address	End Address	Size (Bytes)	Data Access	Address Align-ment	Address Compatibility
Public TIPB Bridge (Continued)						
STROBE0, FFFB:0000 -> FFFB:FFFF (Continued)						
CS = 23 ICR	FFFB:B800	FFFB:BFFF	2K	16	16	P1
CS = 24 MPUIO_1	FFFB:C000	FFFB:C7FF	2K	32	32	None
CS = 25 MPUIO_2	FFFB:C800	FFFB:CFFF	2K	32	32	P1
CS = 26 MPUIO_3	FFFB:D000	FFFB:D7FF	2K	32	32	P1
CS = 27 MPUIO_4	FFFB:D800	FFFB:DFFF	2K	32	32	P1
CS = 28 MPUIO_5	FFFB:E000	FFFB:E7FF	2K	32	32	None
CS = 29 MPUIO_6	FFFB:E800	FFFB:EFFF	2K	32	32	None
CS = 30 SPGPIO_WR	FFFB:F000	FFFB:F7FF	2K			None
CS = 31 Reserved	FFFB:F800	FFFB:FFFF	2K			
STROBE1, FFFC:0000 -> FFFC:FFFF						
CS = 0 LLPC	FFFC:0000	FFFC:07FF	2K	16	16	None
CS = 1 SPI_100K_1	FFFC:0800	FFFC:0FFF	2K	16	16	None
CS = 2 SPI_100K_2	FFFC:1000	FFFC:17FF	2K	16	16	None
CS = 3 SYREN_SPI	FFFC:1800	FFFC:1FFF	2K	16	16	None
CS = 4 HR-TFT LCD	FFFC:2000	FFFC:27FF	2K	16	16	None
CS = 5 to 31 Reserved	FFFC:2800	FFFC:FFFF	54K			
Private TIPB Bridge						
STROBE0, FFFD:0000 -> FFFD:FFFF						
CS = 0 to 31 Reserved	FFFD:0000	FFFD:FFFF	64K			
STROBE1, FFFE:0000 -> FFFE:FFFF						
CS = 0 Level2_INTH	FFFE:0000	FFFE:07FF	2K	32	32	P1, H1, H2
CS = 1 PCC_ULPD	FFFE:0800	FFFE:0FFF	2K	16	32	None
CS = 2 PERSEUS_CONF	FFFE:1000	FFFE:17FF	2K	16	16	P1
CS = 3 GSM_PROTECT	FFFE:1800	FFFE:1FFF	2K	32	32	None
CS = 4 Reserved	FFFE:2000	FFFE:27FF	2K			
CS = 5 Reserved	FFFE:2800	FFFE:2FFF	2K			
CS = 6 Reserved	FFFE:3000	FFFE:37FF	2K			

Table 6–3. TIPB Peripherals Address Space (Continued)

Device Name	Start Address	End Address	Size (Bytes)	Data Access	Address Align-ment	Address Compatibility
Private TIPB Bridge (Continued)						
STROBE1, FFFE:0000 -> FFFE:FFFF (Continued)						
CS = 7 Reserved	FFFE:3800	FFFE:3FFF	2K			
CS = 8 DES/3DES	FFFE:4000	FFFE:47FF	2K	32	32	H2
CS = 9 SHA1/MD5	FFFE:4800	FFFE:4FFF	2K	32	32	H2
CS = 10 RNG	FFFE:5000	FFFE:57FF	2K	32	32	H2
CS = 11 to 20 Unused	FFFE:5800	FFFE:A7FF	20K			
CS = 21 SWATCHDOG	FFFE:A800	FFFE:AFFF	2K	32	32	H2
CS = 22 to 23 Unused	FFFE:B000	FFFE:BFFF	4K			
CS = 24	FFFE:C000	FFFE:C7FF	2K			
LCD	FFFE:C000	FFFE:C0FF	256	32	32	P1, H1, H2
L3 OCP T1	FFFE:C100	FFFE:C1FF	256	32	32	H2
L3 OCP T2	FFFE:C200	FFFE:C2FF	256	32	32	None
L3 OCP initiator	FFFE:C320	FFFE:C3FF	224	32	32	H2
Reserved	FFFE:C400	FFFE:C4FF	256			
Timer1	FFFE:C500	FFFE:C5FF	256	32	32	P1, H1, H2
Timer2	FFFE:C600	FFFE:C6FF	256	32	32	P1, H1, H2
Timer3	FFFE:C700	FFFE:C7FF	256	32	32	P1, H1, H2
CS = 25	FFFE:C800	FFFE:CFFF	2K			
Watchdog timer	FFFE:C800	FFFE:C8FF	256	32	32	P1, H1, H2
Reserved	FFFE:C900	FFFE:C9FF	256			
TIPB bridge1 (private)	FFFE:CA00	FFFE:CAFF	256	16	32	P1, H1, H2
MPU interrupt handler	FFFE:CB00	FFFE:CBFF	256	32	32	P1, H1, H2
Traffic controller	FFFE:CC00	FFFE:CCFF	256	32	32	P1, H1, H2
Reserved	FFFE:CD00	FFFE:CDFF	256			
CLKM	FFFE:CE00	FFFE:CEFF	256	32	32	P1, H1, H2
DPLL1	FFFE:CF00	FFFE:CFFF	256	32	32	P1, H1, H2

Table 6–3. TIPB Peripherals Address Space (Continued)

Device Name	Start Address	End Address	Size (Bytes)	Data Access	Address Alignment	Address Compatibility
Private TIPB Bridge (Continued)						
STROBE1, FFFE:0000 -> FFFE:FFFF (Continued)						
CS = 26	FFFE:D000	FFFE:D7FF	2K			
Reserved	FFFE:D000	FFFE:D0FF	256			
Reserved	FFFE:D100	FFFE:D1FF	256			
DSP MMU	FFFE:D200	FFFE:D2FF	256	32	32	H1, H2
TIPB bridge2 (public)	FFFE:D300	FFFE:D3FF	256	16	32	P1, H1, H2
Test block (PSA)	FFFE:D400	FFFE:D4FF	256	32	32	H2
Reserved	FFFE:D500	FFFE:D7FF				
CS = 27 and 28	FFFE:D800	FFFE:E7FF	4K			
DMA controller	FFFE:D800	FFFE:E7FF	4K	32	32	P1, H1, H2
CS = 29 Reserved	FFFE:E800	FFFE:EFFF	2K			
CS = 30 Reserved	FFFE:F000	FFFE:F7FF	2K			
CS = 31 Reserved	FFFE:F800	FFFE:FFFF	2K			

Chapter 7

MPU-S Interrupt Mapping

This chapter describes MPU-Subsystem (MPU-S) interrupt mapping of the OMAP850 multimedia processor.

Topic **Page**

7.1 MPU-S Interrupt Mapping 7-2

7.1 MPU-S Interrupt Mapping

The interrupt controller handles 32 interrupt lines. This module handles edge-triggered or level-sensitive interrupts. The interrupts are enabled or disabled with an internal register and can be routed on one of the two MPU processor interrupts according to a programmable bit.

The mapping of incoming interrupts is shown in Table 7–1.

Table 7–1. MPU-S Incoming Interrupts

Incoming Interrupts	Default Sensitivity Configuration	Interrupt Line on Level1	Interrupt Line on Level2	Compatibility
Level2 INTH FIQ	Level	IRQ_0		P1
Level2 INTH IRQ	Level	IRQ_1		P1
USB Non-ISO	Level	IRQ_2		P1
USB ISO	Level	IRQ_3		P1
ICR	Edge	IRQ_4		P1
EAC	Level	IRQ_5		P1
MPUIO_1	Edge	IRQ_6		P1
MPUIO_2	Edge	IRQ_7		P1
MPUIO_3	Edge	IRQ_8		P1
IRQ_ABORT (TIPB)	Level	IRQ_9		P1, H1, H2
McBSP2 TX	Edge	IRQ_10		None
McBSP2 RX	Edge	IRQ_11		None
McBSP2 RX overflow	Edge	IRQ_12		None
IRQ_RHEA_BRIDGE_PRIVATE	Level	IRQ_13		P1, H1
IRQ_LCD_LINE	Level	IRQ_14		None
GSM_PROTECT	Level	IRQ_15		None
IRQ_TIMER3	Edge	IRQ_16		P1, H1, H2
MPUIO_5	Edge	IRQ_17		None
MPUIO_6	Edge	IRQ_18		None
IRQ_DMA_CH0	Level	IRQ_19		P1, H1, H2
IRQ_DMA_CH1	Level	IRQ_20		P1, H1, H2

- P1: OMAP710
- P2: OMAP730
- H1: OMAP1510
- H2: OMAP1610

Table 7–1. MPU-S Incoming Interrupts (Continued)

Incoming Interrupts	Default Sensitivity Configuration	Interrupt Line on Level1	Interrupt Line on Level2	Compatibility
IRQ_DMA_CH2	Level	IRQ_21		P1, H1, H2
IRQ_DMA_CH3	Level	IRQ_22		P1, H1, H2
IRQ_DMA_CH4	Level	IRQ_23		P1, H1, H2
IRQ_DMA_CH5	Level	IRQ_24		P1, H1, H2
IRQ_DMA_CH_LCD	Level	IRQ_25		P1, H1, H2
IRQ_TIMER1	Edge	IRQ_26		P1, H1, H2
IRQ_WD_TIMER	Edge	IRQ_27		P1, H1, H2
IRQ_RHEA_BRIDGE_PUBLIC	Level	IRQ_28		P1, H1
SPGIO_WR	Level	IRQ_29		
IRQ_TIMER2	Edge	IRQ_30		P1, H1, H2
IRQ_LCD_CTRL	Level	IRQ_31		P1, H1, H2
HW_errors (TCIF)		IRQ0 or 1	IRQ_00	P1
Fast external power fail interrupt (NFIQ_PWR_FAIL pin)	Level	IRQ0 or 1	IRQ_01	P1
RESERVED		IRQ0 or 1	IRQ_02	P1
RESERVED		IRQ0 or 1	IRQ_03	P1
I^2C	Level	IRQ0 or 1	IRQ_04	H1, H2
PCC	Level	IRQ0 or 1	IRQ_05	None
External interrupt (MPU_EXT_NIRQ pin)	Level	IRQ0 or 1	IRQ_06	P1
SPI_100K_1	Level	IRQ0 or 1	IRQ_07	P1
SYREN_SPI	Level	IRQ0 or 1	IRQ_08	None
VLYNQ	Edge	IRQ0 or 1	IRQ_09	None
MPUIO_4	Edge	IRQ0 or 1	IRQ_10	None
McBSP1 TX	Edge	IRQ0 or 1	IRQ_11	None
McBSP1 RX	Edge	IRQ0 or 1	IRQ_12	None
McBSP1 RX overflow	Edge	IRQ0 or 1	IRQ_13	None

- ❑ P1: OMAP710
- ❑ P2: OMAP730
- ❑ H1: OMAP1510
- ❑ H2: OMAP1610

Table 7–1. MPU-S Incoming Interrupts (Continued)

Incoming Interrupts	Default Sensitivity Configuration	Interrupt Line on Level1	Interrupt Line on Level2	Compatibility
UART_MODEM_IRDA 2	Level	IRQ0 or 1	IRQ_14	P1, H2
UART_MODEM 1	Level	IRQ0 or 1	IRQ_15	P1, H2
MCSI	Level	IRQ0 or 1	IRQ_16	H1, H2
µWIRE TX	Edge	IRQ0 or 1	IRQ_17	None
µWIRE RX	Edge	IRQ0 or 1	IRQ_18	None
SMC CD	Level	IRQ0 or 1	IRQ_19	None
SMC IREQ	Level	IRQ0 or 1	IRQ_20	None
HDQ_1WIRE	Level	IRQ0 or 1	IRQ_21	H2
TIMER32K	Edge	IRQ0 or 1	IRQ_22	P1, H2
MMC_SDIO	Level	IRQ0 or 1	IRQ_23	P1, H1, H2
ULPD	Level	IRQ0 or 1	IRQ_24	P1, H2
RTC synchronous timer	Edge	IRQ0 or 1	IRQ_25	P1, H2
RTC alarm	Level	IRQ0 or 1	IRQ_26	P1, H2
USB HHC 1	Level	IRQ0 or 1	IRQ_27	None
USB HHC 2	Level	IRQ0 or 1	IRQ_28	None
USB geni	Level	IRQ0 or 1	IRQ_29	P1
USB OTG	Level	IRQ0 or 1	IRQ_30	None
Camera IF	Level	IRQ0 or 1	IRQ_31	None
RNG	Level	IRQ0 or 1	IRQ_32	None
DUAL_MODE_TIMER	Level	IRQ0 or 1	IRQ_33	None
DBB_RF_EN	Edge	IRQ0 or 1	IRQ_34	None
ARMIO_Keypad	Edge	IRQ0 or 1	IRQ_35	None
SHA-1/MD5	Level	IRQ0 or 1	IRQ_36	None
SPI_100K_2	Level	IRQ0 or 1	IRQ_37	None
RNG idle mode	Level	IRQ0 or 1	IRQ_38	None
ARMIO_GPIO	Level	IRQ0 or 1	IRQ_39	None

- P1: OMAP710
- P2: OMAP730
- H1: OMAP1510
- H2: OMAP1610

Table 7–1. MPU-S Incoming Interrupts (Continued)

Incoming Interrupts	Default Sensitivity Configuration	Interrupt Line on Level1	Interrupt Line on Level2	Compatibility
LLPC_LCD_CTRL_CAN_BE_OFF	Edge	IRQ0 or 1	IRQ_40	None
LLPC_OE_FALLING_EDGE	Edge	IRQ0 or 1	IRQ_41	None
LLPC_OE_RISING_EDGE	Edge	IRQ0 or 1	IRQ_42	None
LLPC_VSYNC	Edge	IRQ0 or 1	IRQ_43	None
Reserved		IRQ0 or 1	IRQ_40 to IRQ_45	
WAKE_UP_REQ	Level	IRQ0 or 1	IRQ_46	H2
Reserved		IRQ0 or 1	IRQ_47 to IRQ_52	
IRQ_DMA_CH6	Edge	IRQ0 or 1	IRQ_53	H2
IRQ_DMA_CH7	Edge	IRQ0 or 1	IRQ_54	H2
IRQ_DMA_CH8	Edge	IRQ0 or 1	IRQ_55	H2
IRQ_DMA_CH9	Edge	IRQ0 or 1	IRQ_56	H2
IRQ_DMA_CH10	Edge	IRQ0 or 1	IRQ_57	H2
IRQ_DMA_CH11	Edge	IRQ0 or 1	IRQ_58	H2
IRQ_DMA_CH12	Edge	IRQ0 or 1	IRQ_59	H2
IRQ_DMA_CH13	Edge	IRQ0 or 1	IRQ_60	H2
IRQ_DMA_CH14	Edge	IRQ0 or 1	IRQ_61	H2
IRQ_DMA_CH15	Edge	IRQ0 or 1	IRQ_62	H2
NAND flash	Level	IRQ0 or 1	IRQ_63	H2

- P1: OMAP710
- P2: OMAP730
- H1: OMAP1510
- H2: OMAP1610

Chapter 8

MPU-S DMA Requests

This chapter discusses the MPU-Subsystem (MPU-S) direct memory access (DMA) interrupt requests of the OMAP850 multimedia processor.

Topic **Page**

8.1 MPU-S DMA Requests 8-2

8.1 MPU-S DMA Requests

There are three external DMA requests:

- ❑ EXT_DMA_REQ_1 multiplexed with FADD_24 pin (ball L15)
- ❑ EXT_DMA_REQ_2 multiplexed with FADD_14 pin (ball R21)
- ❑ EXT_DMA_REQ_3 multiplexed with FADD_25 pin (ball L18)

You can activate only one of these three requests at a time, as shown in Table 8–1.

Table 8–1. Activated External DMA Requests

Configuration Register	Value	External DMA Request
PERSEUS25_FADD_IOCONF2[22:20]	011	EXT_DMA_REQ_2 activated
PERSEUS2_IO_CONF5[19:17]	xxx	
PERSEUS2_IO_CONF5[23:21]	xxx	
PERSEUS25_FADD_IOCONF2[22:20]	#011	EXT_DMA_REQ_3 activated
PERSEUS2_IO_CONF5[19:17]	100	
PERSEUS2_IO_CONF5[23:21]	xxx	
PERSEUS25_FADD_IOCONF2[22:20]	#011	EXT_DMA_REQ_1 activated
PERSEUS2_IO_CONF5[19:17]	#100	
PERSEUS2_IO_CONF5[23:21]	011	

Table 8–2 lists the MPU-S DMA requests.

Table 8–2. MPU-S DMA Requests

DMA Requests		Configuration	MPU System DMA	Compati-bility
MCSI TX			DMA_REQ_01	H1, H2
MCSI RX	EXT_DMA_REQ	PERSEUS25_MODE[16] = 0/1	DMA_REQ_02	H1, H2
I2C RX	EXT_DMA_REQ	PERSEUS25_MODE[17] = 0/1	DMA_REQ_03	H1, H2
I2C TX			DMA_REQ_04	H1, H2
VLYNQ	EXT_DMA_REQ	PERSEUS25_MODE[18] = 0/1	DMA_REQ_05	None
SHA1/MD5	EXT_DMA_REQ	PERSEUS25_MODE[19] = 0/1	DMA_REQ_06	None
µWIRE TX	EXT_DMA_REQ	PERSEUS25_MODE[20] = 0/1	DMA_REQ_07	H1, H2
McBSP1 TX			DMA_REQ_08	P1, H1, H2
McBSP1 RX	EXT_DMA_REQ	PERSEUS25_MODE[21] = 0/1	DMA_REQ_09	P1, H1, H2
McBSP2 TX			DMA_REQ_10	H1, H2
McBSP2 RX	EXT_DMA_REQ	PERSEUS25_MODE[22] = 0/1	DMA_REQ_11	H1, H2
UART_MODEM 1 TX			DMA_REQ_12	P1, H1, H2

Table 8–2. MPU-S DMA Requests (Continued)

DMA Requests		Configuration	MPU System DMA	Compatibility
UART_MODEM 1 RX	EXT_DMA_REQ	PERSEUS25_MODE[23] = 0/1	DMA_REQ_13	P1, H1, H2
UART_MODEM_IRDA 2 TX	EXT_DMA_REQ	PERSEUS25_MODE[24] = 0/1	DMA_REQ_14	P1, H1, H2
UART_MODEM_IRDA 2 RX			DMA_REQ_15	P1, H1, H2
DES/3DES IN	EXT_DMA_REQ	PERSEUS25_MODE[25] = 0/1	DMA_REQ_16	None
DES/3DES OUT			DMA_REQ_17	None
SMC TX			DMA_REQ_18	None
SMC RX	EXT_DMA_REQ	PERSEUS25_MODE[26] = 0/1	DMA_REQ_19	None
CAMERA_IF RX	EXT_DMA_REQ	PERSEUS25_MODE[27] = 0/1	DMA_REQ_20	H1, H2
MMC_SDIO TX	EXT_DMA_REQ	PERSEUS25_MODE[28] = 0/1	DMA_REQ_21	H1, H2
MMC_SDIO RX			DMA_REQ_22	H1, H2
NAND flash end of burst	EXT_DMA_REQ	PERSEUS25_MODE[29] = 0/1	DMA_REQ_23	H2
EAC REC			DMA_REQ_24	None
EAC PLAY			DMA_REQ_25	None
USB_OTG RX0 (client)			DMA_REQ_26	H1, H2
USB_OTG RX1 (client)			DMA_REQ_27	H1, H2
USB_OTG RX2 (client)	EXT_DMA_REQ	PERSEUS25_MODE[30] = 0/1	DMA_REQ_28	H1, H2
USB_OTG TX0 (client)			DMA_REQ_29	H1, H2
USB_OTG TX1 (client)			DMA_REQ_30	H1, H2
USB_OTG TX2 (client)	EXT_DMA_REQ	PERSEUS25_MODE[31] = 0/1	DMA_REQ_31	H1, H2

Chapter 9

GSM-S Memory Mapping

This chapter describes the memory mapping for and the GSM-Subsystem (GSM-S) of the OMAP850 multimedia processor.

Topic **Page**

9.1	GSM-MPU Memory Mapping	9-2
9.2	External Flash ROM Image	9-3
9.3	GSM-S DSP Memory Space	9-7

9.1 GSM-MPU Memory Mapping

The GSM-MPU memory space is shared between the external memory interface and the TIPB. The memory interface provides six chip-select signals. All internal peripherals are mapped on GSM-MPU memory space with a range of 32K bytes.

The 8K bytes of internal RAM (0380:0000h to 0380:1000h) can overlay the first 8K-byte region 0000:0000h-0000:1000h of the GSM-MPU address space. In this case, the first 8K-byte of the external memory is not accessible by the GSM-MPU. This overlay is controlled by GSM-MPU using a register of GSM-MPU memory interface.

9.1.1 GSM-MPU Memory Mapping

Table 9–1 lists the GSM-MPU memory map.

Table 9–1. GSM-MPU Memory Map

Device Name	nIBOOT	Start Address	Stop Address	Size (Bytes)	Data
nCS0 program†	1	0000:0000	007F:FFFF	8M	8/16/32
	0	0000:2000	007F:FFFF	8M – 8K	
nCS6	-	0080:0000	0084:FFFF	320K	8/16/32
nCS6 DSP-shared	-	0085:0000	0085:FFFF	64K	8/16/32
Not allocated	-	00C0:0000	00FF:FFFF	-	-
nCS1: data†	-	0100:0000	017F:FFFF	8M	8/16/32
nCS2: random†	-	0180:0000	01FF:FFFF	8M	8/16/32
Not allocated	-	0200:0000	02BF:FFFF	-	8/16/32
nCS0 image	-	0300:0000	037F:FFFF	8M	8/16/32
nCS7	1	0380:0000	03FF:FFFF	8M	8/16/32
	0	0000:0000	0000:1FFF	8K	
Debug unit (DU)	-	03C0:0000	03FF:FFFF	32	32
Not allocated	-	0400:0000	FFCF:FFFF	-	-
MPUI RAM	-	FFD0:0000	FFD0:3FFF	16K	16/32
MPUI control register	-	FFE0:0000	FFE0:0001	2	16

† External device

9.2 External Flash ROM Image

Whatever the value of nIBOOT, it is possible to write to or read from (depending on the value of the WE bit) the external memory connected to nCS0 at address range nCS0 image.

Table 9–2 presents the data format.

Table 9–2. Data Format

<table>
<tr><th>D32----------->D24</th><th>D23----------->D16</th><th>D15----------->D8</th><th>D7----------->D0</th></tr>
<tr><td colspan="4">nCS0</td></tr>
<tr><td colspan="4">nCS1</td></tr>
<tr><td colspan="4">nCS2</td></tr>
<tr><td colspan="4">nCS3</td></tr>
<tr><td colspan="4">CS4</td></tr>
<tr><td colspan="4">nCS6</td></tr>
<tr><td colspan="4">nCS7</td></tr>
<tr><td colspan="4">MPUI RAM</td></tr>
<tr><td colspan="2"></td><td colspan="2">MPU</td></tr>
<tr><td colspan="2"></td><td colspan="2">GEA</td></tr>
<tr><td colspan="2" rowspan="13">Not mapped</td><td colspan="2">APIC</td></tr>
<tr><td colspan="2">SIM</td></tr>
<tr><td colspan="2">TSP</td></tr>
<tr><td colspan="2">TPU_REG</td></tr>
<tr><td colspan="2">TPU_RAM</td></tr>
<tr><td>Not mapped</td><td>RTC</td></tr>
<tr><td colspan="2">ULPD</td></tr>
<tr><td>Not mapped</td><td>I2C</td></tr>
<tr><td colspan="2">SPI</td></tr>
<tr><td colspan="2">TIMER1</td></tr>
<tr><td rowspan="4">Not mapped</td><td>LPG</td></tr>
<tr><td>PWL</td></tr>
<tr><td>Reserved</td></tr>
<tr><td>PWT</td></tr>
</table>

Table 9−2. Data Format (Continued)

D32----------->D24	D23----------->D16	D15----------->D8	D7----------->D0
		µWIRE	
		MPUIO	
		Not mapped	Reserved
			UART_MODEM
		TIMER2	
		TIPB bridge	
		INTH	
		Memory interface	
		DMA controller	
		CLKM	
		JTAG ID code	
		Die ID code	

Table 9−3 presents the GSM-MPU peripheral mapping for Strobe 0, and Table 9−4 presents it for Strobe 1.

Table 9−3. GSM-MPU Peripheral Mapping (Strobe 0)

Device Name		Start Address	Stop Address	Size (Bytes)	Data
Reserved	CS0	FFFF:0000	FFFF:07FF		
Reserved	CS1	FFFF:0800	FFFF:0FFF		
TPU registers	CS2	FFFF:1000	FFFF:13FF	1K	16
Reserved	CS3	FFFF:1800	FFFF:1FFF		
Reserved	CS4	FFFF:2000	FFFF:27FF		
Reserved	CS5	FFFF:2800	FFFF:2FFF		
Reserved	CS6	FFFF:3000	FFFF:37FF		
Reserved	CS7	FFFF:3800	FFFF:3FFF		
Reserved	CS8	FFFF:4000	FFFF:47FF		
Reserved	CS9	FFFF:4800	FFFF:4FFF		
Reserved	CS10	FFFF:5000	FFFF:57FF		
UART_MODEM	CS11	FFFF:5800	FFFF:5FFF	2K	8
Reserved	CS12	FFFF:6000	FFFF:67FF		
Reserved	CS13	FFFF:6800	FFFF:6FFF		
RIF	CS14	FFFF:7000	FFFF:77FF	2K	16
Reserved	CS15	FFFF:7800	FFFF:7FFF		

Table 9–3. GSM-MPU Peripheral Mapping (Strobe 0) (Continued)

Device Name		Start Address	Stop Address	Size (Bytes)	Data
Reserved	CS16	FFFF:8000	FFFF:87FF		
Reserved	CS17	FFFF:8800	FFFF:8FFF		
TPU RAM	CS18	FFFF:9000	FFFF:97FF	2K	16
DPLL configuration	CS19	FFFF:9800	FFFF:9801	2	16
Not allocated	CS20	FFFF:A000	FFFF:A7FF		
Not allocated	CS21	FFFF:A800	FFFF:AFFF		
Not allocated	CS22	FFFF:B000	FFFF:B7FF		
Not allocated	CS23	FFFF:B800	FFFF:BFFF		
GEA	CS24	FFFF:C000	FFFF:C7FF	2K	8/16
Not allocated	CS25	FFFF:C800	FFFF:CFFF		
Not allocated	CS26	FFFF:D000	FFFF:D7FF		
Not allocated	CS27	FFFF:D800	FFFF:DFFF		
Not allocated	CS28	FFFF:E000	FFFF:E7FF		
Not allocated	CS29	FFFF:E800	FFFF:EFFF		
Reserved	CS30	FFFF:F000	FFFF:F7FF		
Watchdog timer	CS31	FFFF:F800	FFFF:F8FF	256	16
TIPB bridge		FFFF:F900	FFFF:F9FF	256	16
INTH		FFFF:FA00	FFFF:FAFF	256	16
Memory interface		FFFF:FB00	FFFF:FBFF	256	16
DMA controller		FFFF:FC00	FFFF:FCFF	256	16
CLKM		FFFF:FD00	FFFF:FDFF	256	16
JTAG ID code		FFFF:FE00	FFFF:FE03	4	16
MPU		FFFF:FF00	FFFF:FFFF	256	16

Table 9–4. GSM-MPU Peripheral Mapping (Strobe 1)

Device Name		Start Address	Stop Address	Size (Bytes)	Data
SIM	CS0	FFFE:0000	FFFE:07FF	2K	16
TSP	CS1	FFFE:0800	FFFE:0FFF	2K	16
Reserved	CS2	FFFE:1000	FFFE:17FF		
RTC	CS3	FFFE:1800	FFFE:1FFF	2K	8
ULPD	CS4	FFFE:2000	FFFE:27FF	2K	16

Table 9–4. GSM-MPU Peripheral Mapping (Strobe 1) (Continued)

Device Name		Start Address	Stop Address	Size (Bytes)	Data
I^2C	CS5	FFFE:2800	FFFE:2FFF	2K	8
SPI	CS6	FFFE:3000	FFFE:37FF	2K	16
TIMER1	CS7	FFFE:3800	FFFE:3FFF	2K	16
UWIRE	CS8	FFFE:4000	FFFE:47FF	2K	16
MPUIO	CS9	FFFE:4800	FFFE:4FFF	2K	16
Reserved	CS10	FFFE:5000	FFFE: 57FF		
Reserved	CS11	FFFE:5800	FFFE: 5FFF		
Reserved	CS12	FFFE:6000	FFFE: 67FF		
TIMER2	CS13	FFFE:6800	FFFE:6FFF	2K	16
Reserved	CS14	FFFE:7000	FFFE:77FF		
LPG	CS15	FFFE:7800	FFFE:7FFF	2K	8
PWL	CS16	FFFE:8000	FFFE:87FF	2K	8
PWT	CS17	FFFE:8800	FFFE:8FFF	2K	8
Reserved	CS18	FFFE:9000	FFFE:97FF		
Reserved	CS19	FFFE:9800	FFFE:9FFF		
Not allocated	CS20	FFFE:A000	FFFE:A7FF		
TCIF	CS21	FFFE:A800	FFFE:AFFF	2K	16
ICR	CS22	FFFE:B000	FFFE:B7FF	2K	16
Not allocated	CS23	FFFE:B800	FFFE:BFFF		
Reserved	CS24	FFFE:C000	FFFE:C7FF		
Not allocated	CS25	FFFE:C800	FFFE:CFFF		
Not allocated	CS26	FFFE:D000	FFFE:D7FF		
Not allocated	CS27	FFFE:D800	FFFE:DFFF		
Not allocated	CS28	FFFE:E000	FFFE:E7FF		
Not allocated	CS29	FFFE:E800	FFFE:EFFF		
PERSEUS2_CONF	CS30	FFFE:F000	FFFE:F7FF	2K	16

9.3 GSM-S DSP Memory Space

The DSP core is embedded with 28K words (16 bits) of random access memory and 128K words (16 bits) of read only memory.

The GSM-S DSP memory space consists of the following kinds of memory:

- DARAM: Dual-access data RAM. It is always mapped in data space and can be overlaid in program space using the OVLY bit.
- MPUIRAM: Dual-access data RAM. It is always mapped in data space and can be overlaid in program space using the OVLY bit. The GSM-MPU host processor can also access this memory via the MPUI interface module. It behaves as a communication memory between the lead CPU and the GSM-MPU host processor.
- PROM: Program ROM, always in program space
- DROM: Data ROM, always in data space
- PDROM: Program or data ROM. This ROM is always mapped in program space and can also be mapped in data space by setting the DROM control bit.
- Shared PDRAM: Program/data RAM mapped on both the data space and the program space of the DSP XIO interface

For this S28C128 configuration the memory mapping is as follows:

- 28K words of data memory (RAM based) mapped in both data space 0 and 1.
  - 2K words of dual access memory (DARAM)
  - 8K words of dual access memory (MPUI DARAM) shared between DSP and MPU/DMA
  - 18K words of dual access memory (DARAM)
- 128K words of program memory (ROM based)
  - 100K words of program memory (PROM) mapped in program space 0.
  - 20K words of data memory (DROM) mapped in data space 1.

8K words of mixed program/data memory (PDROM) are mapped in both program space 0 and data space 1.

Table 9–5 shows how the DSP memory is mapped.

Table 9–5. DSP Memory Mapping

	Data	Prog0	Prog1	Prog2	Prog3	Prog4	Prog5	Prog6
0000	DARAM overlay over the program area. 2K							
0800	MPUI overlay over the program area 8K							

Table 9–5. DSP Memory Mapping (Continued)

<table>
<tr><th></th><th colspan="2">Data</th><th>Prog0</th><th>Prog1</th><th>Prog2</th><th>Prog3</th><th>Prog4</th><th>Prog5</th><th>Prog6</th></tr>
<tr><td>1000</td><td colspan="9" rowspan="3"></td></tr>
<tr><td>1800</td></tr>
<tr><td>2000</td></tr>
<tr><td>2800</td><td colspan="9" rowspan="10">DARAM overlay over the program area
18K</td></tr>
<tr><td>3000</td></tr>
<tr><td>3800</td></tr>
<tr><td>4000</td></tr>
<tr><td>4800</td></tr>
<tr><td>5000</td></tr>
<tr><td>5800</td></tr>
<tr><td>6000</td></tr>
<tr><td>6800</td></tr>
<tr><td>7000</td><td colspan="2" rowspan="2"></td><td rowspan="13">PROM
32K</td><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td></tr>
<tr><td>7800</td></tr>
<tr><td>8000</td><td></td><td rowspan="4"></td><td rowspan="12"></td><td rowspan="12"></td><td rowspan="2">PROM
8K</td><td rowspan="12"></td><td rowspan="12"></td><td rowspan="12"></td></tr>
<tr><td>8800</td><td rowspan="11">PDRAM
32K</td></tr>
<tr><td>9000</td><td rowspan="10">DROM
20K</td><td rowspan="10"></td></tr>
<tr><td>9800</td></tr>
<tr><td>A000</td></tr>
<tr><td>A800</td></tr>
<tr><td>B000</td></tr>
<tr><td>B800</td></tr>
</table>

Table 9–5. DSP Memory Mapping (Continued)

	Data		Prog0	Prog1	Prog2	Prog3	Prog4	Prog5	Prog6
C000	DROM M=0	DROM M=1							
C800									
D000									
D800									
E000		PDROM 8K							
E800									
F000									
F800									

9.3.1 MPUI Shared Memory

The MPUI offers dual-access capability to 8K words of 16 bits of mixed data program memory.

The MPUI can be configured to manage data access of 8, 16, or 32 bits through the MPUI control registers and the memory interface configuration registers.

The MPUI dual-access capability is either enabled (SAM) or disabled (HOM) by the DSP. SAM is the default configuration when the DSP exits from a reset phase.

In shared access mode (SAM), the GSM-MPU (or DMA controller) and DSP can access the shared memory space simultaneously, with GSM-MPU access resynchronized on the DSP cycle clock (three+E times ratio required between GSM-MPU and DSP cycle clocks).

In host only mode (HOM), the MPUI RAM is dedicated to the sole external access under the control of either the GSM-MPU or the DMA controller; therefore, the access time is limited by the maximum access time of the DARAM used.

9.3.2 XIO Memory Mapping

All of the data space is mapped on page 0 from address 0x8000 to 0xFFFF.

To avoid overlaying the DROM memory space, the DROM bit is used to select either internal DROM (DROM = 1) or external PDRAM *(*DROM = 0).

The program space is mapped in the extended page 4, one-half-pages of 32K words of 16 bits. This configuration prevents the DSP memory extension from overlaying the lower half page of each extended page allocated to the existing DARAM and MPUIRAM.

Depending on the amount of memory targeted for program execution, the program space is mapped from 0x48000 to 0x4FFFF in page 4.

The sharing of this memory capacity between the DSP and the GSM-MPU can be statically configured through a dedicated register:

- ❑ Case 1 => 0M bits for DSP/0.5M bits for GSM-MPU
- ❑ Case 2 => 0.5M bits for DSP/0M bits for GSM-MPU

9.3.3 XIO-TIPB

Internal and external peripherals are mapped on XIO or data memory spaces. These spaces are accessible through nXSTROBE[3:0] with a range of 2K bytes for external peripherals, allowing connection up to:

- ❑ 6 external devices on program space
- ❑ 26 external devices on data space
- ❑ 31 external devices on I/O space

The 32-bit internal peripherals are directly connected on the internal memory interface.

Table 9–6 describes the DSP XIO memory space.

Table 9–6. DSP XIO Memory Space

Device Name		Start Address	Stop Address	Size in Bytes	Data
External Peripherals Mapping - Program Space					
		Strobe 0			
Not allocated	CS0	0000	07FF	2K	16
…	…	…	…	…	…
Not allocated	CS5	3000	37FF	2K	16
External Peripherals Mapping - Data Space 1					
		Strobe 1			
Not allocated	CS6	3800	3FFF	2K	16
UART_MODEM	CS14	7000	77FF	2K	8
MCSI (map1)	CS15	7800	7FFF	2K	16
External Peripherals Mapping - Data Space 2					
		Strobe 2			
Not allocated	CS16	8000	87FF	2K	16
…	…	…	…	…	…
Not allocated	CS31	F800	FFFF	2K	16

Table 9–6. DSP XIO Memory Space (Continued)

Device Name		Start Address	Stop Address	Size in Bytes	Data
External Peripherals Mapping -I/O Space					
Strobe 3					
RIF	CS0	0000	07FF	2K	16
MCSI-1 (DAI)	CS1	0800	0FFF	2K	16
GPO	CS2	1000	17FF	2K	…
Not allocated	CS3	1800	1FFF	2K	…
Not allocated	CS4	2000	27FF	2K	…
A51/2	CS5	2800	2FFF	2K	16
Not allocated		…	…		…
Not allocated	CS14	7000	77FF	2K	16
Not allocated	CS15	7800	7FFF	2K	16
Not allocated	…	…	…	…	…
DMA controller	CS29	E800	EFFF	2K	16
Not allocated	CS30	F000	F7FF	2K	16
XI0-TIPB bridge	CS31	F800	F8FF	256	16
MPUI control		F900	F9FF	256	16
INTH		FA00	FAFF	256	16
NMI_ST_REG		FB00	FBFF	256	16
Not allocated		FC00	FCFF	256	
Not allocated		FD00	FDFF	256	
Not allocated		FE00	FEFF	256	
Not allocated		FF00	FFFF	256	

Chapter 10

Frame Buffer

This chapter describes the frame buffer of the OMAP850 multimedia processor.

Topic **Page**

10.1 Frame Buffer **10-2**

10.1 Frame Buffer

This module is replaced by a new frame buffer.

Two new configuration bits are added in PERSEUS25_CONF module.

- Frame buffer clock control: For performance purposes, clock autogating is removed. However, the frame buffer clock can be statically shutoff by the FB_CLOCK_EN = PERSEUS25_MODE[1] bit when the frame buffer is not used. The default is enable.
- Frame buffer OCP response delay: Control of frame buffer response delay to OCP command, that is, the default command is *accept* or *no*, via the FB_RESP = PERSEUS25_MODE[2] bit.

Appendix A

Pin Descriptions

This appendix describes the OMAP850 platform pins.

Topic **Page**

A.1 Pin Description by Module A-2

A.2 Pin Multiplexing A-36

A.1 Pin Descriptions by Module

Table A–1 lists the pin descriptions by module.

Table A–1.Pin Descriptions by Module

	2 μA	10 μA	Dual Voltage	Failsafe
Pull-up	PS0201	PS1001	UPS295	
Pull-down	PD0201	PD1001	UPS290	PE0201

Table A–2. TPU/TSP

Signal	Designation	Type	Mode 0					Mode 2					Mode 3					Mode 5				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
TSPACT_0	Synchronous activation signal	O	W12	tspact_0	PS0201	0	1															
TSPACT_1	Synchronous activation signal	O	R12	tspact_1	PS0201	0	2															
TSPACT_2	Synchronous activation signal	O	P12	tspact_2	PS0201	0	3															
TSPACT_3	Synchronous activation signal	O	W11	tspact_3	PS0201	0	4															
TSPACT_4	Synchronous activation signal	O	V11	tspact_4	PS0201	0	5															
TSPACT_5	Synchronous activation signal	O						AA20	smc_io	PS0201	9	2						R8	mpu_uart_tx1	PE0201	3	5
TSPACT_6	Synchronous activation signal	O						V17	smc_clk	PS0201	9	3						Y3	mpu_uart_rx1	PE0201	3	5
TSPACT_7	Synchronous activation signal	O						W19	smc_rst	PE0201	9	4						AA1	mpu_uart_cts1	PE0201	3	6
TSPACT_8	Synchronous activation signal	O						V18	smc_cd	PE0201	9	5						AA2	mpu_uart_rts1	PE0201	3	6
TSPACT_9	Synchronous activation signal	O						Y19	smc_pwctrl	PE0201	9	6	V4	cdi	PE0201	4	6					

Table A–2. TPU/TSP (Continued)

Signal	Designation	Type	Mode 0					Mode 2					Mode 3					Mode 5				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
TSPACT_10	Synchronous activation signal	O											W2	mclk	PE0201	4	7					
TSPACT_11	Synchronous activation signal	O											V18	smc_cd	PE0201	9	5					
TSPCLKX	Serial clock	O	V10	tspclkx	PS0201	0	0															
TSPDO	Output serial data	O	R11	tspdo	PE0201	0	0															
TSPDI	Input serial data	I											F2	kbr_4	PS1001	13	1					
TSPEN_0	Configurable enable triggers	O	AA9	tspen_0	PE0201	1	0															
TSPEN_1	Configurable enable triggers	O	P11	tspen_1	PE0201	0	7															
TSPEN_2	Configurable enable triggers	O	W10	tspen_2	PE0201	0	6															
TSPEN_3	Configurable enable triggers	O											T3	nemu0	PS0201	10	3					
													F3	kbr_3	PS1001	13	0					

Table A–3.GSM–S Voice AuSPI

Signal	Designation	Type	Mode 1				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
VCLKRX	Transmit/receive clock	I	W6	sclk	PE0201	2	1
VDX	Transmit data	O	R9	sdo	PS0201	2	1
VDR	Receive data	I	Y6	sdi	PE0201	2	1
VFSRX	Transmit/receive synchronization	I	Y5	fsync	PE0201	2	1

Table A–4.GSM–S RIF BB

Signal	Designation	Type	Mode 0					Mode 2				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
BFSR	Receive synchro	I	W8	bfsr	PE0201	1	1					
BFSX	Transmit synchro	O	AA7	bfsx	PS0201	1	1					
BDR	Receive data	I	Y8	bdr	PE0201	1	1					
BDX	Transmit data	O	V8	bdx	PS0201	1	1					
BCLKR	Receive clock	I						K15	nfbe_0	UPS290	8	2
BCLKX	Transmit clock	O						K19	nfbe_1	UPS290	8	1

Table A–5. GSM-S SIM

Signal	Designation	Type	Mode 0				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
SIM_RST	SIM reset	O	Y20	sim_rst	PE0201	1	2
SIM_CD	Card detect	I	P15	sim_cd	PE0201	1	4
SIM_IO	Input output signal	I/O	N14	sim_io	PS0201	1	2
SIM_CLK	Output clock	O	W20	sim_clk	PS0201	1	2
SIM_PWR_CTRL	Power control	O	V19	sim_pwrctrl	PS0201	1	3

Table A–6.GSM-S MCSI

Signal	Designation	Type	Mode 1					Mode 4					Mode 5				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
MCSI_TXD	Transmit serial data	O	L1	lcd_pixel_13	PE0201	3	7	W17	usb_vbusi	PE0201	2	7	V2	mpu_spi1_sdo	PS0201	8	5
			C15	cam_exclk	PS0201	11	1										
MCSI_RXD	Receive serial data	I	L3	lcd_pixel_12	PE0201	3	7	V16	mclk_out	PS0201	3	0	T4	mpu_spi1_sdi	PE0201	8	6
			C16	cam_hs	PS0201	11	2										
MCSI_CLK	Bit synchronization clock	I/O	M8	lcd_pixel_14	PE0201	3	7	W16	usb_pu_en	PS0201	2	6	U3	mpu_spi1_sclk	PE0201	8	4
			D17	cam_lclk	PS0201	11	0										
MCSI_FSYNCH	Frame synchronization clock or SS reset	I/O	M7	lcd_pixel_15	PE0201	3	7	W18	creset	PE0201	3	1	V3	mpu_spi1_sen0	PE0201	8	7
			D15	cam_vs	PS0201	11	3										

Table A–7.GSM-S UART

Signal	**Designation**	**Type**	**Mode 2**					**Mode 4**				
			Ball	**Pad Name**	**PU/PD**	**Conf. Reg.**	**Reg. Field**	**Ball**	**Pad Name**	**PU/PD**	**Conf. Reg.**	**Reg. Field**
UART_TX	Transmit data	O	V16	mclk_out	PS0201	3	0	Y4	mpu_UART_tx_ir2	PS0201	3	2
UART_RX	Receive data	I	W18	creset	PE0201	3	1	V5	mpu_uart_rx_ir2	PE0201	3	3
UART_CTS	Clear to send	I	W16	usb_pu_en	PS0201	2	6					
UART_RTS	Request to send	O	W17	usb_vbusi	PE0201	2	7					

Table A–8. GSM-S μWIRE

Signal	Designation	Type	Mode 1					Mode 3					Mode 4				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
SDI	Data in	I	F4	kbc_3	PE0201	13	5	Y3	mpu_uart_rx1	PE0201	3	5	AA20	smc_io	PS0201	9	2
								T4	mpu_spi1_sdi	PE0201	8	6	A21	cam_rstz	PE0201	11	4
SDO	Data out	O	E4	kbc_2	PE0201	13	4	R8	mpu_uart_tx1	PE0201	3	5	W19	smc_rst	PE0201	9	4
								V2	mpu_spi1_sdo	PS0201	8	5	A20	cam_data_1	PE0201	11	6
SCLK	Serial clock	O	E3	kbc_4	PE0201	13	6	AA1	mpu_uart_cts1	PE0201	3	6	V17	smc_clk	PS0201	9	3
								U3	mpu_spi1_sclk	PE0201	8	4	G14	cam_data_0	PS0201	11	5
nSCS1	Chip select	O	F2	kbr_4	PS1001	13	1	AA2	mpu_uart_rts1	PE0201	3	6	V18	smc_cd	PE0201	9	5
								V3	mpu_spi1_sen0	PE0201	8	7	B19	cam_data_2	PE0201	11	7
nSCS2	Chip select	O											Y19	smc_pwctrl	PE0201	9	6
								U4	mpu_spi1_sen1	PS0201	9	0	D15	cam_vs	PS0201	11	3

Table A–9.GSM-S LPG

Signal	Designation	Type	Mode 1				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
GLPG1	LED1 control signal	O	N2	Mux_mode_MLPG1	PE0201	10	1
GLPG2	LED2 control signal	O	N7	arm_boot_mlpg2	PE0201	10	2

Table A–10. GSM-S I²C

Signal	Designation	Type	Mode 2					Mode 3				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
GSM_I2C_SDA	I²C data line	I/O	V6	mpu_i2c_sda		5	0	C2	kbc_0	PE0201	13	2
GSM_I2C_SCK	I²C clock line	I/O	W5	mpu_i2c_sck		5	1	E2	kbr_0	PS1001	12	5

Table A–11. GSM-S GPIO

Signal	Designation	Type	Mode 1					Mode 2					Mode 3					Mode 4					Mode 5				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
IO_GSM_0	Generic GSM IO signals	I/O	W11	tspact_3	PS0201	0	4	A20	cam_data_1	PE0201	11	6						V4	cdi	PE0201	4	6					
IO_GSM_1	Generic GSM IO signals	I/O	V11	tspact_4	PS0201	0	5	B19	cam_data_2	PE0201	11	7						W2	mclk	PE0201	4	7					
IO_GSM_2	Generic GSM IO signals	I/O	M4	rfen	PE0201	1	7	C18	cam_data_3	PE0201	12	0											U4	mpu_spi1_sen1	PS0201	9	0
								P15	sim_cd	PE0201	1	4															
IO_GSM_3	Generic GSM IO signals	I/O	P3	clk_13m_req	PE0201	10	7																W1	mpu_spi1_sen2	PS0201	9	1
IO_GSM_9	Generic GSM IO signals	I/O	W19	smc_rst	PE0201	9	4																				
IO_GSM_12	Generic GSM IO signals	I/O											K18	nfcs_1	UPS290	7	2										
IO_GSM_13	Generic GSM IO signals	I/O											K14	nfcs_2	UPS290	7	1										
IO_GSM_14	Generic GSM IO signals	I/O											P19	fadd_16	PS0201	6	5										
IO_GSM_15	Generic GSM IO signals	I/O											M19	fadd_23	PS0201	5	6										

Table A–12. MPU-S LCD

Signal	Designation	Type	Mode 0					Mode 1				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
LCD_PXL_0	Pixel data out	O	G3	lcd_pixel_0	PE0201	4	3					
LCD_PXL_1	Pixel data out	O	G2	lcd_pixel_1	PE0201	4	2					
LCD_PXL_2	Pixel data out	O	K8	lcd_pixel_2	PS0201	4	2					
LCD_PXL_3	Pixel data out	O	H4	lcd_pixel_3	PS0201	4	2					
LCD_PXL_4	Pixel data out	O	G1	lcd_pixel_4	PS0201	4	2					
LCD_PXL_5	Pixel data out	O	H2	lcd_pixel_5	PS0201	4	2					
LCD_PXL_6	Pixel data out	O	H3	lcd_pixel_6	PS0201	4	2					
LCD_PXL_7	Pixel data out	O	K7	lcd_pixel_7	PS0201	4	2					
LCD_PXL_8	Pixel data out	O	J2	lcd_pixel_8	PS0201	4	2					
LCD_PXL_9	Pixel data out	O	K3	lcd_pixel_9	PS0201	4	2					
LCD_PXL_10	Pixel data out	O	L7	lcd_pixel_10	PS0201	4	1					
LCD_PXL_11	Pixel data out	O	L4	lcd_pixel_11	PE0201	4	0					
LCD_PXL_12	Pixel data out	O	L3	lcd_pixel_12	PE0201	3	7					
LCD_PXL_13	Pixel data out	O	L1	lcd_pixel_13	PE0201	3	7					
LCD_PXL_14	Pixel data out	O	M8	lcd_pixel_14	PE0201	3	7					
LCD_PXL_15	Pixel data out	O	M7	lcd_pixel_15	PE0201	3	7					
LCD_PXL_16	Pixel data out	O						Y4	mpu_UART_tx_ir2	PS0201	3	2
LCD_PXL_17	Pixel data out	O						V5	mpu_uart_rx_ir2	PE0201	3	3
LCD_PCLK	Pixel clock	O	J4	lcd_pclk	PE0201	4	3					
LCD_HSYNC	Horizontal synchro	O	J3	lcd_hsync	PE0201	4	3					
LCD_VSYNC	Vertical synchro	O	K4	lcd_vsync	PE0201	4	4					
LCD_AC	LCD Bias	O	G4	lcd_ac	PE0201	4	3					

Table A–13. MPU-S HR TFT LCD

Signal	Designation	Type	Mode 1					Mode 3					Mode 4					Mode 7				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
HRTFT_PIXEL_0	Pixel data out	O						G3	lcd_pixel_0	PE0201	4	3										
HRTFT_PIXEL_1	Pixel data out	O						G2	lcd_pixel_1	PE0201	4	2										
HRTFT_PIXEL_2	Pixel data out	O						K8	lcd_pixel_2	PS0201	4	2										
HRTFT_PIXEL_3	Pixel data out	O						H4	lcd_pixel_3	PS0201	4	2										

Table A–13. MPU-S HR TFT LCD (Continued)

Signal	Designation	Type	Mode 1					Mode 3					Mode 4					Mode 7				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
HRTFT_PIXEL_4	Pixel data out	O						G1	lcd_pixel_4	PS0201	4	2										
HRTFT_PIXEL_5	Pixel data out	O						H2	lcd_pixel_5	PS0201	4	2										
HRTFT_PIXEL_6	Pixel data out	O						H3	lcd_pixel_6	PS0201	4	2										
HRTFT_PIXEL_7	Pixel data out	O						K7	lcd_pixel_7	PS0201	4	2										
HRTFT_PIXEL_8	Pixel data out	O						J2	lcd_pixel_8	PS0201	4	2										
HRTFT_PIXEL_9	Pixel data out	O						K3	lcd_pixel_9	PS0201	4	2										
HRTFT_ PIXEL_10	Pixel data out	O						L7	lcd_pixel_10	PS0201	4	1										
HRTFT_ PIXEL_11	Pixel data out	O						L4	lcd_pixel_11	PE0201	4	0										
HRTFT_ PIXEL_12	Pixel data out	O						L3	lcd_pixel_12	PE0201	3	7										
HRTFT_ PIXEL_13	Pixel data out	O						L1	lcd_pixel_13	PE0201	3	7										
HRTFT_ PIXEL_14	Pixel data out	O						M8	lcd_pixel_14	PE0201	3	7										
HRTFT_ PIXEL_15	Pixel data out	O						M7	lcd_pixel_15	PE0201	3	7										
LCD_PIXEL_16	Pixel data out	O	Y4	mpu_uart_tx_ir2	PS0201	3	2															
LCD_PIXEL_17	Pixel data out	O	V5	mpu_uart_rx_ir2	PE0201	3	3															
HRTFT_DCLK	Dot clock	O						J4	lcd_pclk	PE0201	4	3										
HRTFT_LP	Line pulse	O						J3	lcd_hsync	PE0201	4	3										
HRTFT_SPS	Start panel sync	O						K4	lcd_vsync	PE0201	4	4										
HRTFT_ENABLE	Enable	O						G4	lcd_ac	PE0201	4	3										
HRTFT_ASC	Analog switch control	O						W13	sdmc_clk	PE0201	2	2						V17	smc_clk	PS0201	9	3
HRTFT_SSC	Source signal control	O						R13	sdmc_cmd	PE0201	2	2						V19	sim_pwrctrl	PS0201	1	3
HRTFT_SPL	Start pulse left	O						V12	sdmc_dat_0	PE0201	2	2						W19	smc_rst	PE0201	9	4
HRTFT_PS	Power saving	O						Y12	sdmc_dat_1	PE0201	2	2						AA20	smc_io	PS0201	9	2
HRTFT_REV	Reversing	O						V13	sdmc_dat_2	PE0201	2	3						V18	smc_cd	PE0201	9	5

Table A–13. MPU-S HR TFT LCD (Continued)

Signal	Designation	Type	Mode 1					Mode 3					Mode 4					Mode 7				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
HRTFT_CLS	Clock and sync	O						Y13	sdmc_dat_3	PE0201	2	4						Y19	smc_pwctrl	PE0201	9	6
HRTFT_CSYNC	Sync	)											J3	lcd_hsync	PE0201	4	3					

Table A–14. MPU-S UART Modem-IrDA

Signal	Designation	Type	Mode 0					Mode 2					Mode 3					Mode 4				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
UART_TX2_IR	Infrared transmit pulse	O	Y4	mpu_UART_tx_ir2	PS0201	3	2	AA17	usb_dm		2	5	L14	fadd_21	PS0201	6	0					
UART_RX2_IR	Infrared receive pulse	I	V5	mpu_uart_rx_ir2	PE0201	3	3	Y17	usb_dp		2	5	M18	fadd_22	PS0201	5	7					
UART_SD2	IrDA transceiver shutdown mode	O	W4	mpu_uart_sd2	PS0201	3	4															
UART_TX2	Transmit data	O						R8	mpu_uart_tx1	PE0201	3	5						M18	fadd_22	PS0201	5	7
UART_RX2	Receive data	I						Y3	mpu_uart_rx1	PE0201	3	5						L14	fadd_21	PS0201	6	0
UART_CTS2	Clear to send	I						AA1	mpu_uart_cts1	PE0201	3	6						M19	fadd_23	PS0201	5	6
UART_RTS2	Request to send	O						AA2	mpu_uart_rts1	PE0201	3	6						L15	fadd_24	PS0201	5	5

Table A–15. MPU-S UART Modem

Signal	Designation	Type	Mode 0					Mode 1					Mode 3				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
UART_TX1	Transmit data	O	R8	mpu_uart_tx1	PE0201	3	5	AA17	usb_dm		2	5					
UART_RX1	Receive data	I	Y3	mpu_uart_rx1	PE0201	3	5	Y17	usb_dp		2	5					
UART_CTS1	Clear to send	I	AA1	mpu_uart_cts1	PE0201	3	6	W16	usb_pu_en	PS0201	2	6					
UART_RTS1	Request to send	O	AA2	mpu_uart_rts1	PE0201	3	6	W17	usb_vbusi	PE0201	2	7					
UART_DCD1	Data carrier detect	I						V16	mclk_out	PS0201	3	0					
UART_DSR1	Data set ready	I						W18	creset	PE0201	3	1					
UART_DTR1	Data transmit ready	O											W1	mpu_spi1_sen2	PS0201	9	1

Table A–16. MPU-S SPI 100K 1

Signal	Designation	Type	Mode 0					Mode 1					Mode 4				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
MPU_SPI1_SCLK	Serial clock	O	U3	mpu_spi1_sclk	PE0201	8	4	W13	sdmc_clk	PE0201	2	2	R8	mpu_uart_tx1	PE0201	3	5
MPU_SPI1_SDO	Output Serial Data	O	V2	mpu_spi1_sdo	PS0201	8	5	R13	sdmc_cmd	PE0201	2	2	Y3	mpu_uart_rx1	PE0201	3	5
MPU_SPI1_SDI	Input Serial Data	I	T4	mpu_spi1_sdi	PE0201	8	6	V12	sdmc_dat_0	PE0201	2	2	AA1	mpu_uart_cts1	PE0201	3	6
MPU_SPI1_SEN0	Chip Select Enable 0	O	V3	mpu_spi1_sen0	PE0201	8	7	Y12	sdmc_dat_1	PE0201	2	2	AA2	mpu_uart_rts1	PE0201	3	6
MPU_SPI1_SEN1	Chip Select Enable 1	O	U4	mpu_spi1_sen1	PS0201	9	0	V13	sdmc_dat_2	PE0201	2	3					
MPU_SPI1_SEN2	Chip Select Enable 2	O	W1	mpu_spi1_sen2	PS0201	9	1	Y13	sdmc_dat_3	PE0201	2	4					

Table A–17. MPU-S SPI 100K 2

Signal	Designation	Type	Mode 2					Mode 4				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
MPU_SPI2_SCLK	Serial clock	O	M7	lcd_pixel_15	PE0201	3	7	N19	fadd_20	PS0201	6	1
								F2	kbr_4	PS1001	13	1
MPU_SPI2_SDO	Output Serial Data	O	M8	lcd_pixel_14	PE0201	3	7	N18	fadd_19	PS0201	6	2
								E4	kbc_2	PE0201	13	4
MPU_SPI2_SDI	Input Serial Data	I	L1	lcd_pixel_13	PE0201	3	7	N20	fadd_18	PS0201	6	3
								F4	kbc_3	PE0201	13	5
MPU_SPI2_SEN0	Chip Select Enable 0	O	L3	lcd_pixel_12	PE0201	3	7	M15	fadd_17	PS0201	6	4
								E3	kbc_4	PE0201	13	6
MPU_SPI2_SEN1	Chip Select Enable 1	O	L4	lcd_pixel_11	PE0201	4	0	P19	fadd_16	PS0201	6	5
								C2	kbc_0	PE0201	13	2
MPU_SPI2_SEN2	Chip Select Enable 2	O	L7	lcd_pixel_10	PS0201	4	1					
								D3	kbc_1	PE0201	13	3

Table A–18. MPU-S MMC/SDIO

Signal	Designation	Type	Mode 0				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
SDMC_CLK	MMC Clock line	O	W13	sdmc_clk	PE0201	2	2
SDMC_CMD	MMC Command line	I/O	R13	sdmc_cmd	PE0201	2	2
SDMC_DAT0	MMC Data line	I/O	V12	sdmc_dat_0	PE0201	2	2
SDMC_DAT1	MMC Data line	I/O	Y12	sdmc_dat_1	PE0201	2	2
SDMC_DAT2	MMC Data line	I/O	V13	sdmc_dat_2	PE0201	2	3
SDMC_DAT3	MMC Data line	I/O	Y13	sdmc_dat_3	PE0201	2	4

Table A–19. MPU-S I²C

Signal	Designation	Type	Mode 0					Mode 1					Mode 3				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
MPU_I2C_SDA	I2C clock line	I/O	V6	mpu_i2c_sda	PS0201	5	0	E1	kbr_2	PS1001	12	7	V19	sim_pwrctrl	PS0201	1	3
MPU_I2C_SCK	I2C data line	I/O	W5	mpu_i2c_sck	PS0201	5	1	F3	kbr_3	PS1001	13	0	P15	sim_cd	PE0201	1	4

Table A–20. MPU-S HDQ-1WIRE

Signal	Designation	Type	Mode 1					Mode 2				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
HDQ1W		I/O	W4	mpu_uart_sd2	PS0201	3	4	T2	nemu1	PS0201	10	4

Table A–21. MPU-S μWIRE

Signal	Designation	Type	Mode 1					Mode 5				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
UW_SCLK	Transmit clock	O	J4	lcd_pclk	PE0201	4	3	V17	smc_clk	PS0201	9	3
UW_SDO	Serial output data	O	J3	lcd_hsync	PE0201	4	3	W19	smc_rst	PE0201	9	4
UW_SDI	Serial receive data	I	G4	lcd_ac	PE0201	4	3	AA20	smc_io	PS0201	9	2
UW_nSCS1	Chip Select 1	O	G3	lcd_pixel_0	PE0201	4	3	V18	smc_cd	PE0201	9	5
UW_nSCS2	Chip Select 2	O	K4	lcd_vsync	PE0201	4	4	Y19	smc_pwctrl	PE0201	9	6

Table A–22. MPU-S PWL

Signal	Designation	Type	Mode 1				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
PWL	Pulse Width Light modulator	O	T2	nemu1	PS0201	10	4

Table A–23. MPU-S PWT

Signal	Designation	Type	Mode 3				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
PWT	Pulse Width Tone modulator	O	T2	nemu1	PS0201	10	4

Table A–24. MPU-S LPG

Signal	Designation	Type	Mode 0				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
MLPG1	LED1 control signal	O	N2	Mux_mode_MLPG1	PE0201	10	1
MLPG2	LED2 control signal	O	N7	arm_boot_mlpg2	PE0201	10	2

Table A–25. MPU-S Extended GPIO

Signal	Designation	Type	Mode 1					Mode 3				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
EXT_IO_0	Extended Generic input/output.	I/O	E2	kbr_0	PS1001	12	5	N2	Mux_mode_MLPG1	PE0201	10	1
EXT_IO_1	Extended Generic input/output.	I/O	J7	kbr_1	PS1001	12	6	N7	arm_boot_mlpg2	PE0201	10	2
EXT_IO_2	Extended Generic input/output.	I/O	V6	mpu_i2c_sda		5	0					
			C2	kbc_0	PE0201	13	2					
EXT_IO_3	Extended Generic input/output.	I/O	W5	mpu_i2c_sck		5	1					
			D3	kbc_1	PE0201	13	3					

Table A–26. MPU-S EAC BT AuSPI

Signal	Designation	Type	Mode 0					Mode 1					Mode 4				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
SCLK	Bluetooth port interface serial clock	I/O	W6	sclk	PE0201	2	1	U3	mpu_spi1_sclk	PE0201	8	4	C18	cam_data_3	PE0201	12	0
SDO	Bluetooth port interface serial data output	O	R9	sdo	PS0201	2	1	V2	mpu_spi1_sdo	PS0201	8	5	D16	cam_data_4	PS0201	12	1
SDI	Bluetooth port interface serial data input	I	Y6	sdi	PE0201	2	1	T4	mpu_spi1_sdi	PE0201	8	6	C17	cam_data_5	PE0201	12	2
FSYNC	Bluetooth port interface frame synchro	I/O	Y5	fsync	PE0201	2	1	V3	mpu_spi1_sen0	PE0201	8	7	B17	cam_data_6	PE0201	12	3
SEN1	Bluetooth port interface chip select enable	O						U4	mpu_spi1_sen1	PS0201	9	0	A17	cam_data_7	PS0201	12	4

Table A–27. MPU-S EAC Audio Codec

Signal	Designation	Type	Mode 0					Mode 2				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
CSCLK	Codec port interface serial clock	I/O	W3	csclk	PS0201	4	5					
CDO	Codec port interface serial data output	O	Y1	cdo	PS0201	4	5					
CDI	Codec port interface serial data input	I	V4	cdi	PE0201	4	6					
CSYNC	Codec port interface frame synchro	I/O	Y2	csync	PS0201	4	5					
CRESET	Asynchronous EAC reset. Active low	I	W18	creset	PE0201	3	1	W1	mpu_spi1_sen2	PS0201	9	1
MCLK	Codec master clock input	I	W2	mclk	PE0201	4	7					
MCLK_OUT	Codec master clock output	O	V16	mclk_out	PS0201	3	0					

Table A–28. MPU-S USB OTG

Signal	Designation	Type	Mode 0					Mode 1					Mode 3					Mode 4					Mode 5				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
USB_DP	USB differential (+) line.	I/O	Y17	usb_dp		2	5																			
USB_DM	USB differential (–) line	I/O	AA17	usb_dm		2	5																			
USB_TXD_VP	External USB transceiver data (+) signal	I/O											Y17	usb_dp		2	5									

Table A–28. MPU-S USB OTG (Continued)

Signal	Designation	Type	Mode 0					Mode 1					Mode 3					Mode 4					Mode 5				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
USB_SEO_VM	External USB transceiver data (–) signal	I/O											AA17	usb_dm		2	5										
USB_TXD	External USB transceiver transmit signal	O																Y17	usb_dp		2	5					
USB_SEO	Drive single ended zero	O																AA17	usb_dm		2	5					
USB_VP	External USB transceiver VP function	I											W18	creset	PE0201	3	1										
USB_VM	External USB transceiver VM function	I											V16	mclk_out	PS0201	3	0										
USB_VBUSI	Power supply of the USB bus	I	W17	usb_vbusi	PE0201	2	7	P4	mpu_ext_nirq	PS1001	9	7															
USB_TXEN	External USB transceiver output enable	O											W17	usb_vbusi	PE0201	2	7										
USB_PU_EN	Pull_up enable	O	W16	usb_pu_en	PS0201	2	6																				
USB_RCV	External USB transceiver received signal	I											W16	usb_pu_en	PS0201	2	6										

Table A–28. MPU-S USB OTG (Continued)

Signal	Designation	Type	Mode 0					Mode 1					Mode 3					Mode 4					Mode 5				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
USB_SUSPEND	External receiver suspend signal	O																					V6	mpu_i2c_sda		5	0
USB_SPEED	External receiver speed signal	O																					W5	mpu_i2c_sck		5	1

Table A–29. MPU-S McBSP1

Signal	Designation	Type	Mode 1					Mode 2					Mode 4				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
CLKRX1	McBSP Rx/Tx clock	I/O	A21	cam_rstz	PE0201	11	4	W3	csclk	PS0201	4	5	U3	mpu_spi1_sclk	PE0201	8	4
DX1	McBSP data transmit	O	G14	cam_data_0	PS0201	11	5	Y1	cdo	PS0201	4	5	V2	mpu_spi1_sdo	PS0201	8	5
DR1	McBSP data receive	I	A20	cam_data_1	PE0201	11	6	V4	cdi	PE0201	4	6	T4	mpu_spi1_sdi	PE0201	8	6
FSRX1	McBSP frame synchro Rx/Tx	I/O	B19	cam_data_2	PE0201	11	7	Y2	csync	PS0201	4	5	V3	mpu_spi1_sen0	PE0201	8	7
CLKS1	McBSP auxiliary clock	I	C18	cam_data_3	PE0201	12	0	W2	mclk	PE0201	4	7	U4	mpu_spi1_sen1	PS0201	9	0

Table A–30. MPU-S McBSP2

Signal	Designation	Type	Mode 2				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
CLKRX2	McBSP Rx/Tx clock	I/O	J4	lcd_pclk	PE0201	4	3
DX2	McBSP data transmit	O	J3	lcd_hsync	PE0201	4	3
DR2	McBSP data receive	I	K4	lcd_vsync	PE0201	4	4
FSRX2	McBSP frame synchro Rx/Tx	I/O	G3	lcd_pixel_0	PE0201	4	3
CLKS2	McBSP auxiliary clock	I	G4	lcd_ac	PE0201	4	3

Table A–31. MPU-S MCSI

Signal	Designation	Type	Mode 1					Mode 2					Mode 4					Mode 5				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
MCSI_CLK	Bit synchronization clock	I/O	D16	cam_data_4	PS0201	12	1	W6	sclk	PE0201	2	1	P20	fadd_15	PS0201	6 F2*	6 6*	W16	usb_pu_en	PS0201	2	6
								U3	mpu_spi1_sclk	PE0201	8	4										
MCSI_FSYNCH	Frame synchronization clock or SS reset	I/O	A17	cam_data_7	PS0201	12	4	Y5	fsync	PE0201	2	1	R20	fsdd_12	UPS290	6 F2*	6 3*	W18	creset	PE0201	3	1
								V3	mpu_spi1_sen0	PE0201	8	7										
MCSI_RXD	Receive serial data	I	B17	cam_data_6	PE0201	12	3	Y6	sdi	PE0201	2	1	P18	fadd_13	PS0201	6 F2*	6 4*	V16	mclk_out	PS0201	3	0
								T4	mpu_spi1_sdi	PE0201	8	6										
MCSI_TXD	Transmit serial data	O	C17	cam_data_5	PE0201	12	2	R9	sdo	PS0201	2	1	R21	fadd_14	PS0201	6 F2*	6 5*	W17	usb_vbusi	PE0201	2	7
								V2	mpu_spi1_sdo	PS0201	8	5										

Note: With PERSEUS25_MODE[3] = 1

Table A–32. MPU-S SDRAM

Signal	Designation	Type	Mode 0				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
SADD_13	SDRAM address bus out	O	H11	sadd_13			
SADD_12	SDRAM address bus out	O	H9	sadd_12			
SADD_11	SDRAM address bus out	O	H10	sadd_11			
SADD_10	SDRAM address bus out	O	B8	sadd_10			
SADD_9	SDRAM address bus out	O	B12	sadd_9			
SADD_8	SDRAM address bus out	O	G9	sadd_8			

Table A–32. MPU-S SDRAM (Continued)

Signal	Designation	Type	Mode 0				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
SADD_7	SDRAM address bus out	O	G11	sadd_7			
SADD_6	SDRAM address bus out	O	G12	sadd_6			
SADD_5	SDRAM address bus out	O	B9	sadd_5			
SADD_4	SDRAM address bus out	O	G10	sadd_4			
SADD_3	SDRAM address bus out	O	A1	sadd_3			
SADD_2	SDRAM address bus out	O	B6	sadd_2			
SADD_1	SDRAM address bus out	O	B2	sadd_1			
SADD_0	SDRAM address bus out	O	A2	sadd_0			
SBANK_1	SDRAM bank select	O	C3	sbank_1			
SBANK_0	SDRAM bank select	O	B3	sbank_0			
SDATA_15	SDRAM data bus	I/O	C12	sdata_15			
SDATA_14	SDRAM data bus	I/O	D12	sdata_14			
SDATA_13	SDRAM data bus	I/O	D13	sdata_13			
SDATA_12	SDRAM data bus	I/O	C11	sdata_12			
SDATA_11	SDRAM data bus	I/O	C13	sdata_11			
SDATA_10	SDRAM data bus	I/O	D11	sdata_10			
SDATA_9	SDRAM data bus	I/O	D14	sdata_9			
SDATA_8	SDRAM data bus	I/O	C10	sdata_8			
SDATA_7	SDRAM data bus	I/O	D8	sdata_7			
SDATA_6	SDRAM data bus	I/O	C4	sdata_6			

Table A–32. MPU-S SDRAM (Continued)

Signal	Designation	Type	Mode 0				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
SDATA_5	SDRAM data bus	I/O	C7	sdata_5			
SDATA_4	SDRAM data bus	I/O	D5	sdata_4			
SDATA_3	SDRAM data bus	I/O	D7	sdata_3			
SDATA_2	SDRAM data bus	I/O	C5	sdata_2			
SDATA_1	SDRAM data bus	I/O	C6	sdata_1			
SDATA_0	SDRAM data bus	I/O	D6	sdata_0			
SDCLK	SDRAM clock	O	C9	sdclk			
NSRAS	SDRAM row address strobe	O	H7	nsras			
NSCAS	SDRAM column address strobe	O	B4	nscas			
NSWE	SDRAM write enable	O	H8	nswe			
NSDQML	SDRAM low data byte mask	O	C8	nsdqml			
NSDQMU	SDRAM upper data byte mask	O	D10	nsdqmu			
nSDCS	SDRAM chip select	O	G8	nsdcs		5	2
SDCLK_EN	SDRAM power down control signal	O	H12	sdclk_en			

Table A–33. MPU-S DDR

Signal	Designation	Type	Mode 0				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
DQSH	Data strobe high	I/O	C14	dqsh		5	3
DQSL	Data strobe low	I/O	D4	dqsl		5	3
SDCLKX	Clock	O	D9	sdclkx		5	3

Table A–34. MPU-S EMIFS

Signal	Designation	Type	Mode 0					Mode 1					Mode 2					Mode 3				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
FDATA_0	Data bus	I/O	J18	fdata_0	UPS290	7	0															
FDATA_1	Data bus	I/O	J20	fdata_1	UPS290	7	0															
FDATA_2	Data bus	I/O	J15	fdata_2	UPS290	7	0															
FDATA_3	Data bus	I/O	H19	fdata_3	UPS290	7	0															
FDATA_4	Data bus	I/O	H18	fdata_4	UPS290	7	0															
FDATA_5	Data bus	I/O	H15	fdata_5	UPS290	7	0															
FDATA_6	Data bus	I/O	G20	fdata_6	UPS290	7	0															
FDATA_7	Data bus	I/O	G19	fdata_7	UPS290	7	0															
FDATA_8	Data bus	I/O	G18	fdata_8	UPS290	7	0															
FDATA_9	Data bus	I/O	F20	fdata_9	UPS290	7	0															
FDATA_10	Data bus	I/O	F19	fdata_10	UPS290	7	0															
FDATA_11	Data bus	I/O	E20	fdata_11	UPS290	7	0															
FDATA_12	Data bus	I/O	E19	fdata_12	UPS290	7	0															
FDATA_13	Data bus	I/O	F18	fdata_13	UPS290	7	0															
FDATA_14	Data bus	I/O	D19	fdata_14	UPS290	7	0															
FDATA_15	Data bus	I/O	E18	fdata_15	UPS290	7	0															
FADD_1	Address bus out	O	W21	fadd_1	UPS290	6 F1*	6 0*															

Note: With PERSEUS25_MODE[3] = 1

Table A–34. MPU-S EMIFS (Continued)

Signal	Designation	Type	Mode 0					Mode 1					Mode 2					Mode 3				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
FADD_2	Address bus out	O	U18	fadd_2	UPS290	6 F1*	6 1*															
FADD_3	Address bus out	O	V20	fadd_3	UPS290	6 F1*	6 2*															
FADD_4	Address bus out	O	U19	fadd_4	UPS290	6 F1*	6 3*															
FADD_5	Address bus out	O	T18	fadd_5	UPS290	6 F1*	6 4*															
FADD_6	Address bus out	O	U20	fadd_6	UPS290	6 F1*	6 5*															
FADD_7	Address bus out	O	N15	fadd_7	UPS290	6 F1*	6 6*															
FADD_8	Address bus out	O	T19	fadd_8	UPS290	6 F1*	6 7*															
FADD_9	Address bus out	O	T20	fadd_9	UPS290	6 F2*	6 0*															
FADD_10	Address bus out	O	R18	fadd_10	PS0201	6 F2*	7 1*															
FADD_11	Address bus out	O	R19	fadd_11	PS0201	6 F2*	6 2*															
FADD_12	Address bus out	O	R20	fadd_12	UPS290	6 F2*	6 3*															
FADD_13	Address bus out	O	P18	fadd_13	PS0201	6 F2*	6 4*															
FADD_14	Address bus out	O	R21	fadd_14	PS0201	6 F2*	6 5*															
FADD_15	Address bus out	O	P20	fadd_15	PS0201	6 F2*	6 6*															
FADD_16	Address bus out	O	P19	fadd_16	PS0201	6	5															
FADD_17	Address bus out	O	M15	fadd_17	PS0201	6	4															
FADD_18	Address bus out	O	N20	fadd_18	PS0201	6	3															

Note: With PERSEUS25_MODE[3] = 1

Table A–34. MPU-S EMIFS (Continued)

Signal	Designation	Type	Mode 0					Mode 1					Mode 2					Mode 3				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
FADD_19	Address bus out	O	N18	fadd_19	PS0201	6	2															
FADD_20	Address bus out	O	N19	fadd_20	PS0201	6	1															
FADD_21	Address bus out	O	L14	fadd_21	PS0201	6	0															
FADD_22	Address bus out	O	M18	fadd_22	PS0201	5	7															
FADD_23	Address bus out	O	M19	fadd_23	PS0201	5	6															
FADD_24	Address bus out	O	L15	fadd_24	PS0201	5	5															
FADD_25	Address bus out	O	L18	fadd_25	PS0201	5	4	K14	nfcs_2	UPS290	7	1										
NFCS_0	Chip selects active low	O											L19	nfadv	UPS290	8	0	K19	nfbe_1	UPS290	8	1
													L18	fadd_25	PS0201	5	4					
NFCS_1	Chip selects active low	O	K18	nfcs_1	UPS290	7	2															
NFCS_2	Chip selects active low	O	K14	nfcs_2	UPS290	7	1															
NFCS_3	Chip selects active low	O	J19	nfcs_3	UPS290	7	3															
NFCS3_H	Chip selects active low	O											K18	nfcs_1	UPS290	7	2					
NFCS3_L	Chip selects active low	O											J19	nfcs_3	UPS290	7	3					
NFWE	Memory write enable	O	C19	nfwe	UPS290	7	4															

Note: With PERSEUS25_MODE[3] = 1

Table A–34. MPU-S EMIFS (Continued)

Signal	Designation	Type	Mode 0					Mode 1					Mode 2					Mode 3				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
NFOE	Data output enable	O	D18	nfoe	UPS290	7	4															
NFWP	Flash write protect	O	B20	nfwp	UPS290	7	4															
NFBAA	Burst address advance for burst Flash	O	C21	nfbaa	UPS290	7	5															
NFRST	Power down for TI/Reset for Intel Flash	O	C20	nfrst	UPS290	8	3															
NFWAIT	Ready for TI/ Wait for Intel Flash	I	B21	nfwait	PS0201	7	6															
NFADV	Address valid	O	L19	nfadv	UPS290	8	0															
FCLK	Clock for burst flash	O	J21	fclk		7	7															
NFBE_0	16-LSB Byte low/high accesses enable	O	K15	nfbe_0	UPS290	8	2															
NFBE_1	16-LSB Byte low/high accesses enable	O	K19	nfbe_1	UPS290	8	1															

Note: With PERSEUS25_MODE[3] = 1

Table A–35. MPU-S NAND Flash

Signal	Designation	Type	Mode 1				
			Ball	Pad Name	PU/PD	Conf. Reg.	REg. Field
I/O_0	In/Out	I/O	T20	fadd_9	UPS290	6 F2*	6 0*
I/O_1	In/Out	I/O	U18	fadd_2	UPS290	6 F1*	6 1*
I/O_2	In/Out	I/O	V20	fadd_3	UPS290	6 F1*	6 2*
I/O_3	In/Out	I/O	U19	fadd_4	UPS290	6 F1*	6 3*
I/O_4	In/Out	I/O	T18	fadd_5	UPS290	6 F1*	6 4*
I/O_5	In/Out	I/O	U20	fadd_6	UPS290	6 F1*	6 5*
I/O_6	In/Out	I/O	N15	fadd_7	UPS290	6 F1*	6 6*
I/O_7	In/Out	I/O	R20	fadd_12	UPS290	6 F2*	6 3*
WE	Write Enable	O	P18	fadd_13	PS0201	6 F2*	6 4*
RE	Read Enable	O	W21	fadd_1	UPS290	6 F1*	6 0*
WP	Write Protect	O	T19	fadd_8	UPS290	6 F1*	6 7*
CLE	Command Latch Enable	O	R21	fadd_14	PS0201	6 F2*	6 5*

Table A–35. MPU-S NAND Flash (Continued)

Signal	Designation	Type	Mode 1				
			Ball	Pad Name	PU/PD	Conf. Reg.	REg. Field
ALE	Address Latch Enable	O	P20	fadd_15	PS0201	6 F2*	6 6*
CE1	Chip Enable 1	O	R19	fadd_11	PS0201	6 F2*	6 2*
CE2	Chip Enable 2	O	R18	fadd_10	PS0201	6 F2*	6 1*
RDY	Ready	I	P19 B21	fadd_16 nfwait	PS0201 PS0201	6 7	5 6

Table A–36. MPU-S Camera

Signal	Designation	Type	Mode 0				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
CAM_EXCLK	Clock output for camera module	O	C15	cam_exclk	PS0201	11	1
CAM_LCLK	Image data latch clock	O	D17	cam_lclk	PS0201	11	0
CAM_HS	Horizontal Sync signal	O	C16	cam_hs	PS0201	11	2
CAM_VS	Vertical Sync signal	O	D15	cam_vs	PS0201	11	3
CAM_RSTZ	Exclusive power supply for serial I/O ports	O	A21	cam_rstz	PE0201	11	4
CAM_DATA_0	Digital Image Data	O	G14	cam_data_0	PS0201	11	5
CAM_DATA_1	Digital Image Data	O	A20	cam_data_1	PE0201	11	6
CAM_DATA_2	Digital Image Data	O	B19	cam_data_2	PE0201	11	7
CAM_DATA_3	Digital Image Data	O	C18	cam_data_3	PE0201	12	0

Table A–36. MPU-S Camera (Continued)

Signal	Designation	Type	Mode 0				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
CAM_DATA_4	Digital Image Data	O	D16	cam_data_4	PS0201	12	1
CAM_DATA_5	Digital Image Data	O	C17	cam_data_5	PE0201	12	2
CAM_DATA_6	Digital Image Data	O	B17	cam_data_6	PE0201	12	3
CAM_DATA_7	Digital Image Data	O	A17	cam_data_7	PS0201	12	4

Table A–37. MPU-S MPUIO

Signal	Designation	Type	Mode 2					Mode 4				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
ARMIO_0	Keypad Row	I/O						D17	cam_lclk	PS0201	11	0
ARMIO_1	Keypad Row	I/O						C15	cam_exclk	PS0201	11	1
ARMIO_2	Keypad Row	I/O						C16	cam_hs	PS0201	11	2
ARMIO_3	Keypad Row	I/O	J7	kbr_1	PS1001	12	6					
ARMIO_4	Keypad Row	I/O	E1	kbr_2	PS1001	12	7					

Table A–38. MPU-S Keypad

Signal	Designation	Type	Mode 0					Mode 1					Mode 2					Mode 3				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
KBC_0	Keypad Column	I	C2	kbc_0	PE0201	13	2															
KBC_1	Keypad Column	I	D3	kbc_1	PE0201	13	3															
KBC_2	Keypad Column	I	E4	kbc_2	PE0201	13	4															
KBC_3	Keypad Column	I	F4	kbc_3	PE0201	13	5															
KBC_4	Keypad Column	I	E3	kbc_4	PE0201	13	6															
KBC_5	Keypad Column	I						Y1	cdo	PS0201	4	5	D15	cam_vs	PS0201	11	3	W19	smc_rst	PE0201	9	4
KBC_6	Keypad Column	I						V4	cdi	PE0201	4	6	A21	cam_rstz	PE0201	11	4	Y19	smc_pwctrl	PE0201	9	6
KBC_7	Keypad Column	I											G14	cam_data_0	PS0201	11	5					
KBR_0	Keypad Row	I	E2	kbr_0	PS1001	12	5															
KBR_1	Keypad Row	I	J7	kbr_1	PS1001	12	6															
KBR_2	Keypad Row	I	E1	kbr_2	PS1001	12	7															
KBR_3	Keypad Row	I	F3	kbr_3	PS1001	13	0															
KBR_4	Keypad Row	I	F2	kbr_4	PS1001	13	1															
KBR_5	Keypad Row	I						Y2	csync	PS0201	4	5	D17	cam_lclk	PS0201	11	0	AA20	smc_io	PS0201	9	2
KBR_6	Keypad Row	I						W3	csclk	PS0201	4	5	C15	cam_exclk	PS0201	11	1	V17	smc_clk	PS0201	9	3
KBR_7	Keypad Row	I											C16	cam_hs	PS0201	11	2					

Table A–39. MPU-S SMC

Signal	Designation	Type	Mode 0				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
SMC_IO	In/Out	I/O	AA20	smc_io	PS0201	9	2
SMC_CLK	Clock	O	V17	smc_clk	PS0201	9	3
SMC_RST	Reset	O	W19	smc_rst	PE0201	9	4
SMC_CD	Card Detect	I	V18	smc_cd	PE0201	9	5
SMC_PWCTRL	Power Control	O	Y19	smc_pwctrl	PE0201	9	6

Table A–40. MPU-S ETM9

Signal	Designation	Type	Mode 3					Mode 4				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
TRACESYNC	Trace Sync Signal		C15	cam_exclk	PS0201	11	1					
TRACESYNCB	Trace Sync Signal							E1	kbr_2	PS1001	12	7
TRACECLK	Trace Clock		D17	cam_lclk	PS0201	11	0					
PIPESTAT_0	Pipeline Status		A21	cam_rstz	PE0201	11	4					
PIPESTAT_1	Pipeline Status		D15	cam_vs	PS0201	11	3					
PIPESTAT_2	Pipeline Status		C16	cam_hs	PS0201	11	2					
PIPESTAT_3	Pipeline Status		P3	clk_13m_req	PE0201	10	7					
PIPESTAT_4	Pipeline Status							J7	kbr_1	PS1001	12	6
PIPESTAT_5	Pipeline Status							E2	kbr_0	PS1001	12	5
TRACEPKT0	Data Trace Packet		G14	cam_data_0	PS0201	11	5					
TRACEPKT1	Data Trace Packet		A20	cam_data_1	PE0201	11	6					
TRACEPKT2	Data Trace Packet		B19	cam_data_2	PE0201	11	7					
TRACEPKT3	Data Trace Packet		C18	cam_data_3	PE0201	12	0					

Table A–40. MPU-S ETM9 (Continued)

Signal	Designation	Type	Mode 3					Mode 4				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
TRACEPKT4	Data Trace Packet		D16	cam_data_4	PS0201	12	1					
TRACEPKT5	Data Trace Packet		C17	cam_data_5	PE0201	12	2					
TRACEPKT6	Data Trace Packet		B17	cam_data_6	PE0201	12	3					
TRACEPKT7	Data Trace Packet		A17	cam_data_7	PS0201	12	4					

Table A–41. MPU-S VLYNQ

Signal	Designation	Type	Mode 1					Mode 2				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
VLYNQ_TXD0	Serial data output	O	R8	mpu_uart_tx1	PE0201	3	5	C17 F3	cam_data_5 kbr_3	PE0201 PS1001	12 13	2 0
VLYNQ_RXD0	Serial data input	I	Y3	mpu_uart_rx1	PE0201	3	5	D16 F2	cam_data_4 kbr_4	PS0201 PS1001	12 13	1 1
VLYNQ_RXD1	Serial data output	O	AA1	mpu_uart_cts1	PE0201	3	6	A17 E3	cam_data_7 kbc_4	PS0201 PE0201	12 13	4 6
VLYNQ_TXD1	Serial data input	I	AA2	mpu_uart_rts1	PE0201	3	6	B17 F4	cam_data_6 kbc_3	PE0201 PE0201	12 13	3 5
VLYNQ_CLK	Serial clock output	O	W2	mclk	PE0201	4	7					

Table A–42. MPU-S SSPI

Signal	Designation	Type	Mode 0				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
MCUDI	Serial data input	I	Y7	mcudi	PE0201	2	0
MCUDO	Serial data output	O	W7	mcudo	PS0201	2	0
MCUEN_0	Device enable	O	V7	mcuen	PS0201	2	0

Table A–43. MPU-S TAP

Signal	Designation	Type	Mode 0					Mode 2					Mode 5				
			Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
TDI	JTAG data input	I	R4	tdi	PE0201												
TDO	JTAG data output	O	R2	tdo	0												
TMS	JTAG mode select	I	R3	tms	PE0201												
TRST	JTAG test reset (active low).	I	U2	ntrst	PE0201												
TCK	JTAG Clock	I	P7	tck	PE0201												
NBSCAN	Boundary-scan selection	I	N4	nbscan	PS0201												
NEMU0	Test emulation pin 0	I/O	T3	nemu0	PS0201	10	3										
NEMU1	Test emulation pin 1	I/O	T2	nemu1	PS0201	10	4										
EXTERN0_MPU	ARM9TDMI Extern0 HW breakpoint	I											F3	kbr_3	PS1001	13	0
EXTERN1_MPU	ARM9TDMI Extern1 HW breakpoint	I											F2	kbr_4	PS1001	13	1
TEST_MODE	Test Mode	I	N3	test_mode	PE0201	10	5										
RTCK	Return JTAG Clock	O						N7	arm_boot_mlpg2	PE0201	10	2					

Table A–44. Miscellaneous

Signal	Designation	Type	Mode	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
CLKTCXO	VCTXO input clock (13 MHz)	I	0	Y9	clktcxo			
CLK13M_OUT	CLKM output clock (13 MHz)	O	0	N1	clk13m_out			
OSC32K_IN	Input component signal of 32 kHz quartz	I	0	AA15	osc32k_in			
OSC32K_OUT	Output component signal of 32 kHz quartz	O	0	W15	osc32k_out			
ON_nOFF	Regulators activity	I	0	R14	on_noff			
NRESPWRON	Chip power-on reset	I	0	V15	nrespwron			
MPU_NRST	MPU reset	I	0	P2	mpu_nrst	PS1001		

Note: With PERSEUS25_MODE[3] = 1.

Table A–44. Miscellaneous (Continued)

Signal	Designation	Type	Mode	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
IT_WAKEUP	Wake-up interrupt of real time clock	O	0	W14	it_wakeup		1	5
TCXOEN	TCXO enable	O	0	R10	tcxoen	PE0201	1	6
RFEN	External RF IC enable	O	0	M4	rfen	PE0201	1	7
MPU_EXT_NIRQ	External MPU IRQ	I	0	P4	mpu_ext_nirq	PS1001	9	7
GSM_EXT_NIRQ	External GSM IRQ	I	0	M3	gsm_ext_nirq	PS1001	10	0
MUX_MODE_MLPG1		I	0	N2	Mux_mode_MLPG1	PE0201	10	1
ARM_BOOT_MLPG2		I	0	N7	arm_boot_mlpg2	PE0201	10	2
EXT_ARM_NIRQ		I	1	AA20	smc_io	PS0201	9	2
EXT_DSP_NIRQ		I	1	V17	smc_clk	PS0201	9	3
IT_FRAME			1	V18	smc_cd	PE0201	9	5
XF			1	Y19	smc_pwctrl	PE0201	9	6
LOW_POWER			1	T3	nemu0	PS0201	10	3
NFIQ_PWRFAIL			4	N2	Mux_mode_MLPG1	PE0201	10	1
CLK32K	Digital 32 kHz output	O	0	V14	clk32k		10	6
CLK48M_IN	CLKM input clock (48 MHz)	I	2	T3	nemu0	PS0201	10	3
CLK13M_IN	CLKM input clock (13 MHz)	I	1	W1	mpu_spi1_sen2	PS0201	9	1
			5	W2	mclk	PE0201	4	7
CLK_13M_REQ	Clock 13 MHz request	I	0	P3	clk_13m_req	PE0201	10	7
			1	L18	fadd_25	PS0201	5	4
EXT_DMA_REQ_1		I	3	L15	fadd_24	PS0201	5	5

Note: With PERSEUS25_MODE[3] = 1.

Table A–44. Miscellaneous (Continued)

Signal	Designation	Type	Mode	Ball	Pad Name	PU/PD	Conf. Reg.	Reg. Field
EXT_DMA_REQ_2		I	3	R21	fadd_14	PS0201	6 F2*	6 5*
EXT_DMA_REQ_3		I	4	L18	fadd_25	PS0201	5	4

Note: With PERSEUS25_MODE[3] = 1.

Table A–45. Powers

Pin	Description	Balls
VDDOMAP	OMAP logic	A3 F2
VDD	OMAP730 logic	B18 H20 AA19 AA3 U1 D2 M3 B1 A9
VDDDLL	DDR DLL	A15
VDDLMM	DBB DSP	B16 D20
VDDGSM	DBB Logic	A13 G21 L21
VDDRTC	RTC Logic	Y14
VDDLDO1	Output of embedded LDO for OMAP DPLLs	K2 A11
VDDLDO2	Output of embedded LDO for DBB DPLL	K20
VPP	eFuse supply	AA21
VDDA	Slicer and APLLs	W9
VDDSHV1	Sdram interface	A5 A7 B10 B14 M20
VDDSHV2	Camera interface	AA5
VDDSHV3	Flash interface	E21 J1 U21
VDDSHV4	SIM/SMC interface	Y21 A19

Table A–45. Powers (Continued)

Pin	Description	Balls
VDDSHV5	RTC backup IO	Y16
VDDSHV6	Mmc interface	AA13
VDDSHV7	USB interface	Y18
VDDSHV8	Global I/Os	R1
VDDSHV9	LCD interface	C1
VDDSHV10	Tspact interface	Y10
VSS	Common ground	B5 B7 B13 G13 H13 H14 J14 M14 P14 P13 P10 P9 P8 N8 L8 J8
VSSA	Analog ground	V9
VSS32K	Ground for OSC32k	Y15
D_VDD	DAGON VDD Core	M2 AA11 N21 B15

A.2 Pin Multiplexing

Note: Changes are in orange shading; reset mode is in yellow shading.

Table A–46. OMAP780 Pin Multiplexing

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field† Other	Reg/ Field‡
V10	IO	TPU/TSP PORT	TSPCLKX											MPU GPIO	GPIO_0			0.0	
W12	IO	TPU/TSP PORT	TSPACT_0											MPU GPIO	GPIO_1			0.1	
R12	IO	TPU/TSP PORT	TSPACT_1											MPU GPIO	GPIO_2			0.2	
P12	IO	TPU/TSP PORT	TSPACT_2											MPU GPIO	GPIO_3			0.3	
W11	IO	TPU/TSP PORT	TSPACT_3	GSM GPIO	IO_GSM_0									MPU GPIO	GPIO_4			0.4	
V11	IO	TPU/TSP PORT	TSPACT_4	GSM GPIO	IO_GSM_1									MPU GPIO	GPIO_5			0.5	
R11	IO	TPU/TSP PORT	TSPDO											MPU GPIO	GPIO_6			0.0	
W10	IO	TPU/TSP PORT	TSPEN_2											MPU GPIO	GPIO_7			0.6	
P11	IO	TPU/TSP PORT	TSPEN_1											MPU GPIO	GPIO_8			0.7	
AA9	IO	TPU/TSP PORT	TSPEN_0											MPU GPIO	GPIO_9			1.0	
W8	IO	GSM BB I/F	BFSR											MPU GPIO	GPIO_10			1.1	
Y8	I	GSM BB I/F	BDR											MPU GPIO	GPIN_1				
AA7	IO	GSM BB I/F	BFSX											MPU GPIO	GPIO_11				
V8	IO	GSM BB I/F	BDX											MPU GPIO	GPIO_12				
N14	IO	GSM SIM	SIM_IO											MPU GPIO	GPIO_13			1.2	
W20	IO	GSM SIM	SIM_CLK											MPU GPIO	GPIO_14				
V19	IO	GSM SIM	SIM_PWRCTRL					MPU I2C	MPU_I2C_SDA					MPU GPIO	GPIO_15	HR–TFT	hrtft_ssc	1.3	
P15	IO	GSM SIM	SIM_CD			GSM GPIO	IO_GSM_2	MPU I2C	MPU_I2C_SCK					MPU GPIO	GPIO_16			1.4	

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
Y20	IO	GSM SIM	SIM_RST											MPU GPIO	GPIO_17			1.2	
W14	O	Power mngt	IT_WA-KEUP															1.5	
R10	IO	CLK mngt	TCXOEN											MPU GPIO	GPIO_18			1.6	
M4	IO	CLK mngt	RFEN	GSM GPIO	IO_GSM_2									MPU GPIO	GPIO_19			1.7	
R14	I	Power mngt	ON_nOFF																
Y7	I	Syren SPI	MCUDI											MPU GPIO	GPIN_2			2.0	
W7	IO	Syren SPI	MCUDO											MPU GPIO	GPIO_20				
V7	IO	Syren SPI	MCUEN											MPU GPIO	GPIO_21				
W6	IO	EAC BT auSPI port	SCLK	GSM Voice I/F	VCLKRX	MPU MCSI	MPU_MCSI_CLK							MPU GPIO	GPIO_22			2.1	
R9	IO	EAC BT auSPI port	SDO	GSM Voice I/F	VDX	MPU MCSI	MPU_MCSI_TXD							MPU GPIO	GPIO_23				
Y6	I	EAC BT auSPI port	SDI	GSM Voice I/F	VDR	MPU MCSI	MPU_MCSI_RXD							MPU GPIO	GPIN_3				
Y5	IO	EAC BT auSPI port	FSYNC	GSM Voice I/F	VFSRX	MPU MCSI	MPU_MCSI_FSYNCH							MPU GPIO	GPIO_24				
W13	IO	MMC/SDIO	SDMC_CLK	MPU SPI_100K_1	MPU_SPI1_SCLK			HR_TFT	Hrtft_asc					MPU GPIO	GPIO_25			2.2	
R13	IO	MMC/SDIO	SDMC_CMD	MPU SPI_100K_1	MPU_SPI1_SDO			HR_TFT	Hrtft_ssc					MPU GPIO	GPIO_26				
V12	IO	MMC/SDIO	SDMC_DAT_0	MPU SPI_100K_1	MPU_SPI1_SDI			HR_TFT	Hrtft_spl			DA-GON_UART	DG_UART_TX	MPU GPIO	GPIO_27				
Y12	IO	MMC/SDIO	SDMC_DAT_1	MPU SPI_100K_1	MPU_SPI1_SEN0			HR_TFT	Hrtft_ps			DA-GON_UART	DG_UART_RX	MPU GPIO	GPIO_28				
V13	IO	MMC/SDIO	SDMC_DAT_2	MPU SPI_100K_1	MPU_SPI1_SEN1			HR_TFT	Hrtft_rev			DA-GON_UART	DG_UART_CTS	MPU GPIO	GPIO_29			2.3	

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
Y13	IO	MMC/SDIO	SDMC_ DAT_3	MPU SPI_ 100K_1	MPU_ SPI1_ SEN2			HR_TFT	Hrtft_cls			DA-GON_UAR T	DG_UART _RTS	MPU GPIO	GPIO_30			2.4	
AA17	IO	USB	USB_DM	MPU UART_ MODEM	MPU_ UART_ TX1	MPU UART_ MODEM_ IRDA	MPU_UAR T_TX_IR2	USB OTG	USB_ SEO_VM	USB	USB_SEO			MPU GPIO	GPIO_31			2.5	
Y17	IO	USB	USB_DP	MPU UART_ MODEM	MPU_ UART_ RX1	MPU UART_ MODEM_ IRDA	MPU_UAR T_RX_IR2	USB OTG	USB_ TXD_VP	USB	USB_TXD			MPU GPIO	GPIO_32				
W16	IO	USB	USB_PU_ EN	MPU UART_ MODEM	MPU_ UART_ CTS1	GSM UART	GSM_ UART_CT S	USB OTG	USB_RCV	GSM MCSI	GSM_ MCSI_CLK	MPU MCSI	MPU_MC SI_CLK	MPU GPIO	GPIO_33			2.6	
W17	IO	USB	USB_VBU-SI	MPU UART_ MODEM	MPU_ UART_ RTS1	GSM UART	GSM_ UART_RT S	USB OTG	USB_ TXEN	GSM MCSI	GSM_ MCSI_ TXD	MPU MCSI	MPU_MC SI_TXD	MPU GPIO	GPIO_34			2.7	
V16	IO	EAC	MCLK_OUT	MPU UART_ MODEM	MPU_ UART_ DCD1	GSM UART	GSM_ UART_TX	USB	USB_VM	GSM MCSI	GSM_ MCSI_ RXD	MPU MCSI	MPU_MC SI_RXD	MPU GPIO	GPIO_35			3.0	
W18	IO	EAC Audio Codec	CRESET	MPU UART_ MODEM	MPU_ UART_ DSR1	GSM UART	GSM_UAR T_RX	USB	USB_VP	GSM MCSI	GSM_ MCSI_ FSYNCH	MPU MCSI	MPU_MC SI_FSYNC H	MPU GPIO	GPIO_36			3.1	
Y4	IO	MPU UART_ MO-DEM_IRDA	MPU_ UART_TX_ IR2	LCD I/F	LCD_ PIXEL_16					GSM UART	GSM_ UART_TX			MPU GPIO	GPIO_37			3.2	
V5	IO	MPU UART_ MO-DEM_IRDA	MPU_ UART_RX_ IR2	LCD I/F	LCD_ PIXEL_17					GSM UART	GSM_ UART_RX			MPU GPIO	GPIO_38			3.3	
W4	IO	MPU UART_ MO-DEM_IRDA	MPU_ UART_SD2	HDQ1wire	HDQ1W									MPU GPIO	GPIO_39			3.4	

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
R8	IO	MPU UART_ MODEM	MPU_ UART_TX1	VLYNQ	VLYNQ_ TXD0	MPU UART_ MODEM_ IRDA	MPU_UAR T_TX2	GSM UWIRE	GSM_ UW_SDO	MPU SPI_ 100K_1	MPU_ SPI1_ SCLK	TPU/TSP PORT	TSPACT_5	MPU GPIO	GPIO_40			3.5	
Y3	IO	MPU UART_ MODEM	MPU_ UART_RX1	VLYNQ	VLYNQ_ RXD0	MPU UART_ MODEM_ IRDA	MPU_UAR T_RX2	GSM UWIRE	GSM_ UW_SDI	MPU SPI_ 100K_1	MPU_ SPI1_SDO	TPU/TSP PORT	TSPACT_6	MPU GPIO	GPIO_41				
AA1	IO	MPU UART_ MODEM	MPU_ UART_ CTS1	VLYNQ	VLYNQ_ RXD1	MPU UART_ MODEM_ IRDA	MPU_UAR T_CTS2	GSM UWIRE	GSM_ UW_SCLK	MPU SPI_ 100K_1	MPU_ SPI1_SDI	TPU/TSP PORT	TSPACT_7	MPU GPIO	GPIO_42			3.6	
AA2	IO	MPU UART_ MODEM	MPU_ UART_ RTS1	VLYNQ	VLYNQ_ TXD1	MPU UART_ MODEM_ IRDA	MPU_UAR T_RTS2	GSM UWIRE	GSM_ UW_ nSCS1	MPU SPI_ 100K_1	MPU_ SPI1_ SEN0	TPU/TSP PORT	TSPACT_8	MPU GPIO	GPIO_43				
M7	IO	LCD I/F	LCD_ PIXEL_15	GSM MCSI	GSM_ MCSI_ FSYNCH	MPU SPI_ 100K_2	MPU_SPI2 _SCLK	HR_TFT	Hrtft_ pixel_15			DA-GON_DE-BUG	DA-GON_OBS _15	MPU GPIO	GPIO_44			3.7	
M8	IO	LCD I/F	LCD_ PIXEL_14	GSM MCSI	GSM_MC SI_CLK	MPU SPI_ 100K_2	MPU_SPI2 _SDO	HR_TFT	Hrtft_ pixel_14			DA-GON_DE-BUG	DA-GON_OBS _14	MPU GPIO	GPIO_45				
L1	IO	LCD I/F	LCD_ PIXEL_13	GSM MCSI	GSM_MC SI_TXD	MPU SPI_ 100K_2	MPU_SPI2 _SDI	HR_TFT	Hrtft_ pixel_13			DA-GON_DE-BUG	DA-GON_OBS _13	MPU GPIO	GPIO_46				
L3	IO	LCD I/F	LCD_ PIXEL_12	GSM MCSI	GSM_MC SI_RXD	MPU SPI_ 100K_2	MPU_SPI2 _SEN0	HR_TFT	Hrtft_ pixel_12			DA-GON_DE-BUG	DA-GON_OBS _12	MPU GPIO	GPIO_47				
L4	IO	LCD I/F	LCD_ PIXEL_11			MPU SPI_ 100K_2	MPU_SPI2 _SEN1	HR_TFT	Hrtft_ pixel_11			DA-GON_DE-BUG	DA-GON_OBS _11	MPU GPIO	GPIO_48			4.0	
L7	IO	LCD I/F	LCD_ PIXEL_10			MPU SPI_ 100K_2	MPU_SPI2 _SEN2	HR_TFT	Hrtft_ pixel_10			DA-GON_DE-BUG	DA-GON_OBS _10	MPU GPIO	GPIO_49			4.1	

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
K3	IO	LCD I/F	LCD_ PIXEL_9					HR_TFT	Hrtft_ pixel_9			DA-GON_DE-BUG	DA-GON_OBS_9	MPU GPIO	GPIO_50			4.2	
J2	IO	LCD I/F	LCD_ PIXEL_8					HR_TFT	Hrtft_ pixel_8			DA-GON_DE-BUG	DA-GON_OBS_8	MPU GPIO	GPIO_51				
K7	IO	LCD I/F	LCD_ PIXEL_7					HR_TFT	Hrtft_ pixel_7			DA-GON_DE-BUG	DA-GON_OBS_7	MPU GPIO	GPIO_52				
H3	IO	LCD I/F	LCD_ PIXEL_6					HR_TFT	Hrtft_ pixel_6			DA-GON_DE-BUG	DA-GON_OBS_6	MPU GPIO	GPIO_53				
H2	IO	LCD I/F	LCD_ PIXEL_5					HR_TFT	Hrtft_ pixel_5			DA-GON_DE-BUG	DA-GON_OBS_5	MPU GPIO	GPIO_54				
G1	IO	LCD I/F	LCD_ PIXEL_4					HR_TFT	Hrtft_ pixel_4			DA-GON_DE-BUG	DA-GON_OBS_4	MPU GPIO	GPIO_55				
H4	IO	LCD I/F	LCD_ PIXEL_3					HR_TFT	Hrtft_ pixel_3			DA-GON_DE-BUG	DA-GON_OBS_3	MPU GPIO	GPIO_56				
K8	IO	LCD I/F	LCD_ PIXEL_2					HR_TFT	Hrtft_ pixel_2			DA-GON_DE-BUG	DA-GON_OBS_2	MPU GPIO	GPIO_57				
G2	IO	LCD I/F	LCD_ PIXEL_1					HR_TFT	Hrtft_ pixel_1			DA-GON_DE-BUG	DA-GON_OBS_1	MPU GPIO	GPIO_58				
G3	IO	LCD I/F	LCD_ PIXEL_0	MPU UWIRE	MPU_UW_nSCS1	MCBSP2	FSRX2	HR_TFT	Hrtft_ pixel_0			DA-GON_DE-BUG	DA-GON_OBS_0	MPU GPIO	GPIO_59			4.3	
J4	IO	LCD I/F	LCD_PCLK	MPU UWIRE	MPU_UW_SCLK	MCBSP2	CLKRX2	HR_TFT	Hrtft_dclk			DA-GON_DE-BUG	DA-GON_OBS_19	MPU GPIO	GPIO_60				
J3	IO	LCD I/F	LCD_ HSYNC	MPU UWIRE	MPU_UW_SDO	MCBSP2	DX2	HR_TFT	Hrtft_lp	HR_TFT	Hrtft_csync	DA-GON_DE-BUG	DA-GON_OBS_18	MPU GPIO	GPIO_61				
K4	IO	LCD I/F	LCD_ VSYNC	MPU UWIRE	MPU_UW_nSCS2	MCBSP2	DR2	HR_TFT	Hrtft_sps			DA-GON_DE-BUG	DA-GON_OBS_17	MPU GPIO	GPIO_62			4.4	

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
G4	IO	LCD I/F	LCD_AC	MPU UWIRE	MPU_UW _SDI	MCBSP2	CLKS2	HR_TFT	Hrtft_ enable			DA-GON_DE-BUG	DA-GON_OBS _16	MPU GPIO	GPIO_63			4.3	
Y2	IO	EAC Audio Codec	CSYNC	Keypad	KBR_5	MCBSP1	FSRX1							MPU GPIO	GPIO_64			4.5	
W3	IO	EAC Audio Codec	CSCLK	Keypad	KBR_6	MCBSP1	CLKRX1							MPU GPIO	GPIO_65				
Y1	IO	EAC Audio Codec	CDO	Keypad	KBC_5	MCBSP1	DX1							MPU GPIO	GPIO_66				
V4	IO	EAC Audio Codec	CDI	Keypad	KBC_6	MCBSP1	DR1	TPU/TSP PORT	TSPACT_9	GSM GPIO	IO_GSM_0			MPU GPIO	GPIO_67			4.6	
W2	IO	EAC Audio Codec	MCLK	VLYNQ	VLYNQ_ CLK	MCBSP1	CLKS1	TPU/TSP PORT	TSPACT_ 10	GSM GPIO	IO_GSM_1	Backup	CLK13M_I N	MPU GPIO	GPIO_68			4.7	
V6	IO	MPU I2C	MPU_I2C_ SDA	Extended GPIO	EXT_IO_2	GSM I2C	GSM_I2C_ SDA					USB	USB_SUS-PEND	MPU GPIO	GPIO_69			5.0	
W5	IO	MPU I2C	MPU_I2C_ SCK	Extended GPIO	EXT_IO_3	GSM I2C	GSM_I2C_ SCK					USB	USB_SPE ED	MPU GPIO	GPIO_70			5.1	
H11	O	SDRAM	SADD_13																
H9	O	SDRAM	SADD_12																
H10	O	SDRAM	SADD_11																
B8	O	SDRAM	SADD_10																
B12	O	SDRAM	SADD_9																
G9	O	SDRAM	SADD_8																
G11	O	SDRAM	SADD_7																
G12	O	SDRAM	SADD_6																
B9	O	SDRAM	SADD_5																
G10	O	SDRAM	SADD_4																
A1	O	SDRAM	SADD_3																

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
B6	O	SDRAM	SADD_2																
B2	O	SDRAM	SADD_1																
A2	O	SDRAM	SADD_0																
C3	O	SDRAM	SBANK_1																
B3	O	SDRAM	SBANK_0																
C12	IO	SDRAM	SDATA_15																
D12	IO	SDRAM	SDATA_14																
D13	IO	SDRAM	SDATA_13																
C11	IO	SDRAM	SDATA_12																
C13	IO	SDRAM	SDATA_11																
D11	IO	SDRAM	SDATA_10																
D14	IO	SDRAM	SDATA_9																
C10	IO	SDRAM	SDATA_8																
D8	IO	SDRAM	SDATA_7																
C4	IO	SDRAM	SDATA_6																
C7	IO	SDRAM	SDATA_5																
D5	IO	SDRAM	SDATA_4																
D7	IO	SDRAM	SDATA_3																
C5	IO	SDRAM	SDATA_2																
C6	IO	SDRAM	SDATA_1																
D6	IO	SDRAM	SDATA_0																

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
C9	IO	SDRAM	SDCLK																
H7	O	SDRAM	NSRAS																
B4	O	SDRAM	NSCAS																
H8	O	SDRAM	NSWE																
C8	O	SDRAM	NSDQML																
D10	O	SDRAM	NSDQMU																
G8	IO	SDRAM	nSDCS											MPU GPIO	GPIO_71			5.2	
H12	O	SDRAM	SDCLK_EN																
C14	IO	DDR	DQSH															5.3	
D4	IO	DDR	DQSL																
D9	IO	DDR	SDCLKX																
L18	IO	EMIF slow	FADD_25	Clocks	clk_13m_ req	Slow memory	NFCS_0			EXT_ DMA_RE Q	ext_dma_ req_3			MPU GPIO	GPIO_75			5.4	
L15	IO	EMIF slow	FADD_24					EXT_ DMA_REQ	ext_dma_ req_1	MPU UART_ MODEM_ IRDA	MPU_ UART_ RTS2			MPU GPIO	GPIO_76			5.5	
M19	IO	EMIF slow	FADD_23					GSM_ GPIO	IO_GSM_ 15	MPU UART_ MODEM_ IRDA	MPU_ UART_ CTS2			MPU GPIO	GPIO_77			5.6	
M18	IO	EMIF slow	FADD_22					MPU UART_ MODEM_ IRDA	MPU_ UART_ RX_IR2	MPU UART_ MODEM_ IRDA	MPU_ UART_ TX2			MPU GPIO	GPIO_78			5.7	
L14	IO	EMIF slow	FADD_21					MPU UART_ MODEM_ IRDA	MPU_ UART_ TX_IR2	MPU UART_ MODEM_ IRDA	MPU_ UART_ RX2			MPU GPIO	GPIO_79			6.0	
N19	IO	EMIF slow	FADD_20							MPU SPI_ 100K_2	MPU_ SPI2_ SCLK			MPU GPIO	GPIO_80			6.1	

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
N18	IO	EMIF slow	FADD_19							MPU SPI_ 100K_2	MPU_ SPI2_SDO			MPU GPIO	GPIO_81			6.2	
N20	IO	EMIF slow	FADD_18							MPU SPI_ 100K_2	MPU_ SPI2_SDI			MPU GPIO	GPIO_82			6.3	
M15	IO	EMIF slow	FADD_17							MPU SPI_ 100K_2	MPU_ SPI2_ SEN0			MPU GPIO	GPIO_83			6.4	
P19	IO	EMIF slow	FADD_16	Hardware_ NFC	RDY			GSM_ GPIO	IO_GSM_ 14	MPU SPI_ 100K_2	MPU_ SPI2_ SEN1			MPU GPIO	GPIO_84			6.5	
P20	IO	EMIF slow	FADD_15	Hardware_ NFC	ALE					MPU_ MCSI	MPU_ MCSI_CLK			MPU GPIO	GPIO_85			6.6	F2 6*
R21	IO	EMIF slow	FADD_14	Hardware_ NFC	CLE			EXT_ DMA_REQ	ext_dma_ req_2	MPU_ MCSI	MPU_ MCSI_ TXD			MPU GPIO	GPIO_86				F2 5*
P18	IO	EMIF slow	FADD_13	Hardware_ NFC	WE					MPU_ MCSI	MPU_ MCSI_ RXD			MPU GPIO	GPIO_87				F2 4*
R20	IO	EMIF slow	FADD_12		I/O_7					MPU_ MCSI	MPU_ MCSI_ FSYNCH			MPU GPIO	GPIO_88				F2 3*
R19	IO	EMIF slow	FADD_11	Hardware_ NFC	CE_1									MPU GPIO	GPIO_89				F2 2*
R18	IO	EMIF slow	FADD_10	Hardware_ NFC	CE_2									MPU GPIO	GPIO_90			6.7	F2 1*

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
T20	IO	EMIF slow	FADD_9		I/O_0									MPU GPIO	GPIO_91			6.6	F2 0*
T19	IO	EMIF slow	FADD_8	Hardware_ NFC	WP									MPU GPIO	GPIO_92				F1 7*
N15	IO	EMIF slow	FADD_7		I/O_6									MPU GPIO	GPIO_93				F1 6*
U20	IO	EMIF slow	FADD_6		I/O_5									MPU GPIO	GPIO_94				F1 5*
T18	IO	EMIF slow	FADD_5		I/O_4									MPU GPIO	GPIO_95				F1 4*
U19	IO	EMIF slow	FADD_4		I/O_3									MPU GPIO	GPIO_96				F1 3*
V20	IO	EMIF slow	FADD_3		I/O_2									MPU GPIO	GPIO_97				F1 2*
U18	IO	EMIF slow	FADD_2		I/O_1									MPU GPIO	GPIO_98				F1 1*
W21	IO	EMIF slow	FADD_1	Hardware_ NFC	RE									MPU GPIO	GPIO_99				F1 0*

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
E18	IO	EMIF slow	FDATA_15											MPU GPIO	GPIO_100			7.0	
D19	IO	EMIF slow	FDATA_14											MPU GPIO	GPIO_101				
F18	IO	EMIF slow	FDATA_13											MPU GPIO	GPIO_102				
E19	IO	EMIF slow	FDATA_12											MPU GPIO	GPIO_103				
E20	IO	EMIF slow	FDATA_11											MPU GPIO	GPIO_104				
F19	IO	EMIF slow	FDATA_10											MPU GPIO	GPIO_105				
F20	IO	EMIF slow	FDATA_9											MPU GPIO	GPIO_106				
G18	IO	EMIF slow	FDATA_8											MPU GPIO	GPIO_107				
G19	IO	EMIF slow	FDATA_7											MPU GPIO	GPIO_108				
G20	IO	EMIF slow	FDATA_6											MPU GPIO	GPIO_109				
H15	IO	EMIF slow	FDATA_5											MPU GPIO	GPIO_110				
H18	IO	EMIF slow	FDATA_4											MPU GPIO	GPIO_111				
H19	IO	EMIF slow	FDATA_3											MPU GPIO	GPIO_112				
J15	IO	EMIF slow	FDATA_2											MPU GPIO	GPIO_113				
J20	IO	EMIF slow	FDATA_1											MPU GPIO	GPIO_114				
J18	IO	EMIF slow	FDATA_0											MPU GPIO	GPIO_115				
K14	IO	EMIF slow	NFCS_2	EMIF slow	FADD_25			GSM_ GPIO	IO_GSM_ 13					MPU GPIO	GPIO_116			7.1	
K18	IO	EMIF slow	NFCS_1			EMIF slow	NFCS3H	GSM_ GPIO	IO_GSM_ 12					MPU GPIO	GPIO_117			7.2	
J19	IO	EMIF slow	NFCS_3			EMIF slow	NFCS3L							MPU GPIO	GPIO_118			7.3	
C19	IO	EMIF slow	NFWE											MPU GPIO	GPIO_119			7.4	
D18	IO	EMIF slow	NFOE											MPU GPIO	GPIO_120				
C21	IO	EMIF slow	NFBAA											MPU GPIO	GPIO_121			7.5	
B20	IO	EMIF slow	NFWP											MPU GPIO	GPIO_122			7.4	
B21	IO	EMIF slow	NFWAIT	Hardware_ NFC	RDY									MPU GPIO	GPIO_123			7.6	

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
J21	IO	EMIF slow	FCLK											MPU GPIO	GPIO_124			7.7	
L19	IO	EMIF slow	NFADV			Slow memory	NFCS_0							MPU GPIO	GPIO_125			8.0	
K19	IO	EMIF slow	NFBE_1			GSM RIF BB I/F	BCLKX	Slow memory	NFCS_0					MPU GPIO	GPIO_126			8.1	
K15	IO	EMIF slow	NFBE_0			GSM RIF BB I/F	BCLKR							MPU GPIO	GPIO_127			8.2	
C20	IO	EMIF slow	NFRST											MPU GPIO	GPIO_128			8.3	
U3	IO	MPU SPI_100K_ 1	MPU_SPI1_ SCLK	EAC BT auSPI port	SCLK	MPU MCSI	MPU_MC SI_CLK	GSM UWIRE	GSM_ UW_SCLK	MCBSP1	CLKRX1	GSM MCSI	GSM_MC SI_CLK	MPU GPIO	GPIO_129			8.4	
V2	IO	MPU SPI_100K_ 1	MPU_SPI1_ SDO	EAC BT auSPI port	SDO	MPU MCSI	MPU_MC SI_TXD	GSM UWIRE	GSM_ UW_SDO	MCBSP1	DX1	GSM MCSI	GSM_MC SI_TXD	MPU GPIO	GPIO_130			8.5	
T4	IO	MPU SPI_100K_ 1	MPU_SPI1_ SDI	EAC BT auSPI port	SDI	MPU MCSI	MPU_MC SI_RXD	GSM UWIRE	GSM_ UW_SDI	MCBSP1	DR1	GSM MCSI	GSM_MC SI_RXD	MPU GPIO	GPIO_131			8.6	
V3	IO	MPU SPI_100K_ 1	MPU_SPI1_ SEN0	EAC BT auSPI port	FSYNC	MPU MCSI	MPU_MC SI_FSYNC H	GSM UWIRE	GSM_ UW_ nSCS1	MCBSP1	FSRX1	GSM MCSI	GSM_MC SI_FSYNC H	MPU GPIO	GPIO_132			8.7	
U4	IO	MPU SPI_100K_ 1	MPU_SPI1_ SEN1	EAC BT auSPI port	SEN1			GSM UWIRE	GSM_ UW_ nSCS2	MCBSP1	CLKS1	GSM GPIO	IO_GSM_2	MPU GPIO	GPIO_133			9.0	
W1	IO	MPU SPI_100K_ 1	MPU_SPI1_ SEN2	Backup	CLK13M_ IN	EAC Audio Codec	CRESET	MPU UART	MPU_ UART_ DTR1			GSM GPIO	IO_GSM_3	MPU GPIO	GPIO_134			9.1	
AA20	IO	SMC	SMC_IO	External MCU IRQ (IN)	EXT_ ARM_ NIRQ	TPU/TSP PORT	TSPACT_5	Keypad	KBR_5	GSM UWIRE	GSM_ UW_SDI	MPU UWIRE	MPU_UW _SDI	MPU GPIO	GPIO_135	HR–TFT	Hrtft_ps	9.2	
V17	IO	SMC	SMC_CLK	External DSP IRQ (IN)	EXT_DSP _NIRQ	TPU/TSP PORT	TSPACT_6	Keypad	KBR_6	GSM UWIRE	GSM_ UW_SCLK	MPU UWIRE	MPU_UW _SCLK	MPU GPIO	GPIO_136	HR–TFT	Hrtft_asc	9.3	
W19	IO	SMC	SMC_RST	GSM GPIO	IO_GSM_9	TPU/TSP PORT	TSPACT_7	Keypad	KBC_5	GSM UWIRE	GSM_ UW_SDO	MPU UWIRE	MPU_UW _SDO	MPU GPIO	GPIO_137	HR–TFT	Hrtft_spl	9.4	
V18	IO	SMC	SMC_CD	TPU IT TDMA (OUT)	IT_FRAME	TPU/TSP PORT	TSPACT_8	TPU/TSP PORT	TSPACT_ 11	GSM UWIRE	GSM_ UW_ nSCS1	MPU UWIRE	MPU_UW _nSCS1	MPU GPIO	GPIO_138	HR–TFT	Hrtft_rev	9.5	

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
Y19	IO	SMC	SMC_ PWCTRL	DSP GPO	XF	TPU/TSP PORT	TSPACT_9	Keypad	KBC_6	GSM UWIRE	GSM_ UW_ nSCS2	MPU UWIRE	MPU_UW _nSCS2	MPU GPIO	GPIO_139	HR–TFT	Hrtft_cls	9.6	
V15	I	System	NRESP-WRON																
P2	I	System	MPU_NRST																
P4	I	System	MPU_EXT_ NIRQ	USB	USB_ VBUSI									MPU GPIO	GPIN_4			9.7	
M3	I	System	GSM_EXT_ NIRQ											MPU GPIO	GPIN_5			10.0	
N2	IO	Config boot/ MPU LPG	Mux_mode_ MLPG1	GSM LPG	GSM_ LPG1			Extended GPIO	EXT_IO_0	System	NFIQ_PWR FAIL			MPU GPIO	GPIO_140			10.1	
N7	IO	Config boot / MPU LPG	ARM_boot_ MLPG2	GSM LPG	GSM_ LPG2	TAP	RTCK	Extended GPIO	EXT_IO_1					MPU GPIO	GPIO_141			10.2	
R4	I	TAP	TDI																
U2	I	TAP	nTRST																
R2	O	TAP	TDO																
R3	I	TAP	TMS																
P7	I	TAP	TCK																
T3	IO	Test & Emulation	NEMU0	System	Low_ power	CLOCKS	CLK48M_I N	TPU/TSP PORT	TSPEN_3					MPU GPIO	GPIO_142			10.3	
T2	IO	Test & Emulation	NEMU1	Light level control	PWL	HDQ1wire	HDQ1W	Buzzer	PWT					MPU GPIO	GPIO_143			10.4	
N4	I	Test & Emulation	NBSCAN																
N3	IO	Test & Emulation	TEST_ MODE											MPU GPIO	GPIN_144			10.5	
Y9	I	Clocks	CLKTCXO																
N1	O	Clocks	CLK13M_ OUT																
AA15	I	Clocks	OSC32K_IN																

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
W15	O	Clocks	OSC32K_ OUT																
V14	O	Clocks	CLK32K															10.6	
P3	IO	Clocks	CLK_13M_ REQ	GSM GPIO	IO_GSM_3			ETM	PIPES-TAT_3					MPU GPIO	GPIO_145			10.7	
D17	IO	CAMERA IF	CAM_LCLK	GSM MCSI	GSM_ MCSI_CL K	Keypad	KBR_5	ETM	TRA-CECLK	ARMIO	armio_0	DEBUG	DE-BUG_12	MPU GPIO	GPIO_146			11.0	
C15	IO	CAMERA IF	CAM_ EXCLK	GSM MCSI	GSM_ MCSI_ TXD	Keypad	KBR_6	ETM	TRACE-SYNC	ARMIO	armio_1	DEBUG	DE-BUG_11	MPU GPIO	GPIO_147			11.1	
C16	IO	CAMERA IF	CAM_HS	GSM MCSI	GSM_ MCSI_ RXD	Keypad	KBR_7	ETM	PIPES-TAT_2	ARMIO	armio_2	DEBUG	DE-BUG_10	MPU GPIO	GPIO_148			11.2	
D15	IO	CAMERA IF	CAM_VS	GSM MCSI	GSM_ MCSI_ FSYNCH	Keypad	KBC_5	ETM	PIPES-TAT_1	GSM UWIRE	GSM_ UW_ nSCS2	DEBUG	DEBUG_9	MPU GPIO	GPIO_149			11.3	
A21	IO	CAMERA IF	CAM_RSTZ	MCBSP1	CLKRX1	Keypad	KBC_6	ETM	PIPES-TAT_0	GSM UWIRE	GSM_ UW_SDI	DEBUG	DEBUG_8	MPU GPIO	GPIO_150			11.4	
G14	IO	CAMERA IF	CAM_ DATA_0	MCBSP1	DX1	Keypad	KBC_7	ETM	TRA-CEPKT_0	GSM UWIRE	GSM_ UW_SCLK	DEBUG	DEBUG_7	MPU GPIO	GPIO_151			11.5	
A20	IO	CAMERA IF	CAM_ DATA_1	MCBSP1	DR1	GSM GPIO	IO_GSM_0	ETM	TRA-CEPKT_1	GSM UWIRE	GSM_ UW_SDO	DEBUG	DEBUG_6	MPU GPIO	GPIO_152			11.6	
B19	IO	CAMERA IF	CAM_ DATA_2	MCBSP1	FSRX1	GSM GPIO	IO_GSM_1	ETM	TRA-CEPKT_2	GSM UWIRE	GSM_ UW_ nSCS1	DEBUG	DEBUG_5	MPU GPIO	GPIO_153			11.7	
C18	IO	CAMERA IF	CAM_ DATA_3	MCBSP1	CLKS1	GSM GPIO	IO_GSM_2	ETM	TRA-CEPKT_3	EAC BT auSPI port	SCLK	DEBUG	DEBUG_4	MPU GPIO	GPIO_154			12.0	
D16	IO	CAMERA IF	CAM_ DATA_4	MPU MCSI	MPU_ MCSI_CL K	VLYNQ	VLYNQ_ RXD0	ETM	TRA-CEPKT_4	EAC BT auSPI port	SDO	DEBUG	DEBUG_3	MPU GPIO	GPIO_155			12.1	
C17	IO	CAMERA IF	CAM_ DATA_5	MPU MCSI	MPU_ MCSI_ TXD	VLYNQ	VLYNQ_ TXD0	ETM	TRA-CEPKT_5	EAC BT auSPI port	SDI	DEBUG	DEBUG_2	MPU GPIO	GPIO_156			12.2	
B17	IO	CAMERA IF	CAM_ DATA_6	MPU MCSI	MPU_ MCSI_ RXD	VLYNQ	VLYNQ_ RXD1	ETM	TRA-CEPKT_6	EAC BT auSPI port	FSYNC	DEBUG	DEBUG_1	MPU GPIO	GPIO_157			12.3	

Table A–46. OMAP780 Pin Multiplexing (Continued)

Ball	Type	Pin not muxed or Mode 0 (Primary Function)		Mode 1 Function		Mode 2 Function		Mode 3 Function		Mode 4 Function		Mode 5 function		Mode 6 function (MPU GPIO)		Mode 7 function		Reg/ Field†Other	Reg/ Field‡
A17	IO	CAMERA IF	CAM_ DATA_7	MPU MCSI	MPU_ MCSI_ FSYNCH	VLYNQ	VLYNQ_ TXD1	ETM	TRA-CEPKT_7	EAC BT auSPI port	SEN1	DEBUG	DEBUG_0	MPU GPIO	GPIO_158			12.4	
E2	IO	Keypad	KBR_0	Extended GPIO	EXT_IO_0			GSM I2C	GSM_I2C_ SCK	ETM	PIPES-TAT_5			MPU GPIO	GPIO_159			12.5	
J7	IO	Keypad	KBR_1	Extended GPIO	EXT_IO_1	ARMIO	armio_3			ETM	PIPES-TAT_4			MPU GPIO	GPIO_160			12.6	
E1	IO	Keypad	KBR_2	MPU I2C	MPU_I2C_ SDA	ARMIO	armio_4			ETM	TRACE-SYNCB			MPU GPIO	GPIO_161			12.7	
F3	IO	Keypad	KBR_3	MPU I2C	MPU_I2C_ SCK	VLYNQ	VLYNQ_ TXD0	TPU/TSP PORT	TSPEN_3			Test & Emulation	EX-TERN0_M PU	MPU GPIO	GPIO_162			13.0	
D2	IO	Keypad	KBR_4	GSM UWIRE	GSM_ UW_ nSCS1	VLYNQ	VLYNQ_ RXD0	TPU/TSP PORT	TSPDI	MPU SPI_ 100K_2	MPU_ SPI2_ SCLK	Test & Emulation	EX-TERN1_M PU	MPU GPIO	GPIO_163			13.1	
C2	IO	Keypad	KBC_0	Extended GPIO	EXT_IO_2			GSM I2C	GSM_I2C_ SDA	MPU SPI_ 100K_2	MPU_ SPI2_ SEN1			MPU GPIO	GPIO_164			13.2	
D3	IO	Keypad	KBC_1	Extended GPIO	EXT_IO_3					MPU SPI_ 100K_2	MPU_ SPI2_ SEN2			MPU GPIO	GPIO_165			13.3	
E4	IO	Keypad	KBC_2	GSM UWIRE	GSM_ UW_SDO					MPU SPI_ 100K_2	MPU_ SPI2_SDO			MPU GPIO	GPIO_166			13.4	
F4	IO	Keypad	KBC_3	GSM UWIRE	GSM_ UW_SDI	VLYNQ	VLYNQ_ RXD1			MPU SPI_ 100K_2	MPU_ SPI2_SDI			MPU GPIO	GPIO_167			13.5	
E3	IO	Keypad	KBC_4	GSM UWIRE	GSM_ UW_SCLK	VLYNQ	VLYNQ_ TXD1			MPU SPI_ 100K_2	MPU_ SPI2_ SEN0			MPU GPIO	GPIO_168			13.6	

Table A–46. OMAP780 Pin Multiplexing (Continued)

† In the ”Conf. Reg.” column:

0	→	PERSEUS2_IO_CONF0
1	→	PERSEUS2_IO_CONF1
2	→	PERSEUS2_IO_CONF2
3	→	PERSEUS2_IO_CONF3
4	→	PERSEUS2_IO_CONF4
5	→	PERSEUS2_IO_CONF5
6	→	PERSEUS2_IO_CONF6
7	→	PERSEUS2_IO_CONF7
8	→	PERSEUS2_IO_CONF8
9	→	PERSEUS2_IO_CONF9
10	→	PERSEUS2_IO_CONF10
11	→	PERSEUS2_IO_CONF11
12	→	PERSEUS2_IO_CONF12
13	→	PERSEUS2_IO_CONF13

In the ”Reg. Field” column:

0	→	Bits 3:1
1	→	Bits 7:5
2	→	Bits 11:9
3	→	Bits 15:13
4	→	Bits 19:17
5	→	Bits 23:21
6	→	Bits 27:25
7	→	Bits 31:29

‡ For the EMIFS address multiplexing (), if PERSEUS25_MODE[3] = 1:

In the ”Conf. Reg.” column:

F1	→	PERSEUS25_FADD_IOCONF1
F2	→	PERSEUS25_FADD_IOCONF2

In the ”Reg. Field” column:

0	→	Bits 2:0
1	→	Bits 6:4
2	→	Bits 10:8
3	→	Bits 14:12
4	→	Bits 18:16
5	→	Bits 22:20
6	→	Bits 26:24
7	→	Bits 30:28

Appendix B

Packaging

This appendix presents the OMAP850 package information and mechanical data.

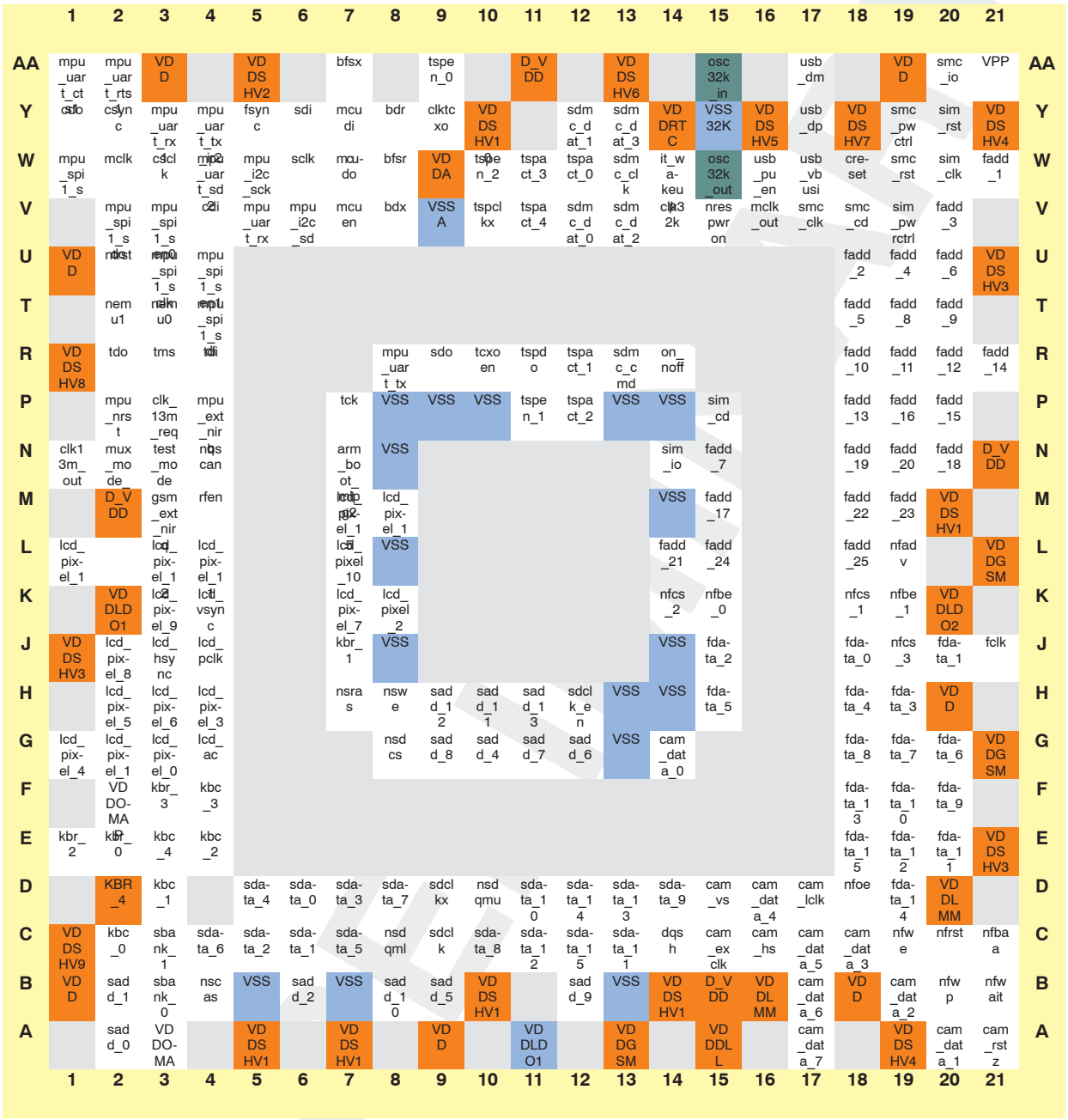
Topic **Page**

B.1 Package Pin Location B-2

B.2 Mechanical Data B-3

B.1 Package Pin Location

Figure B–1. OMAP850 Package Pin Location (Bottom View)

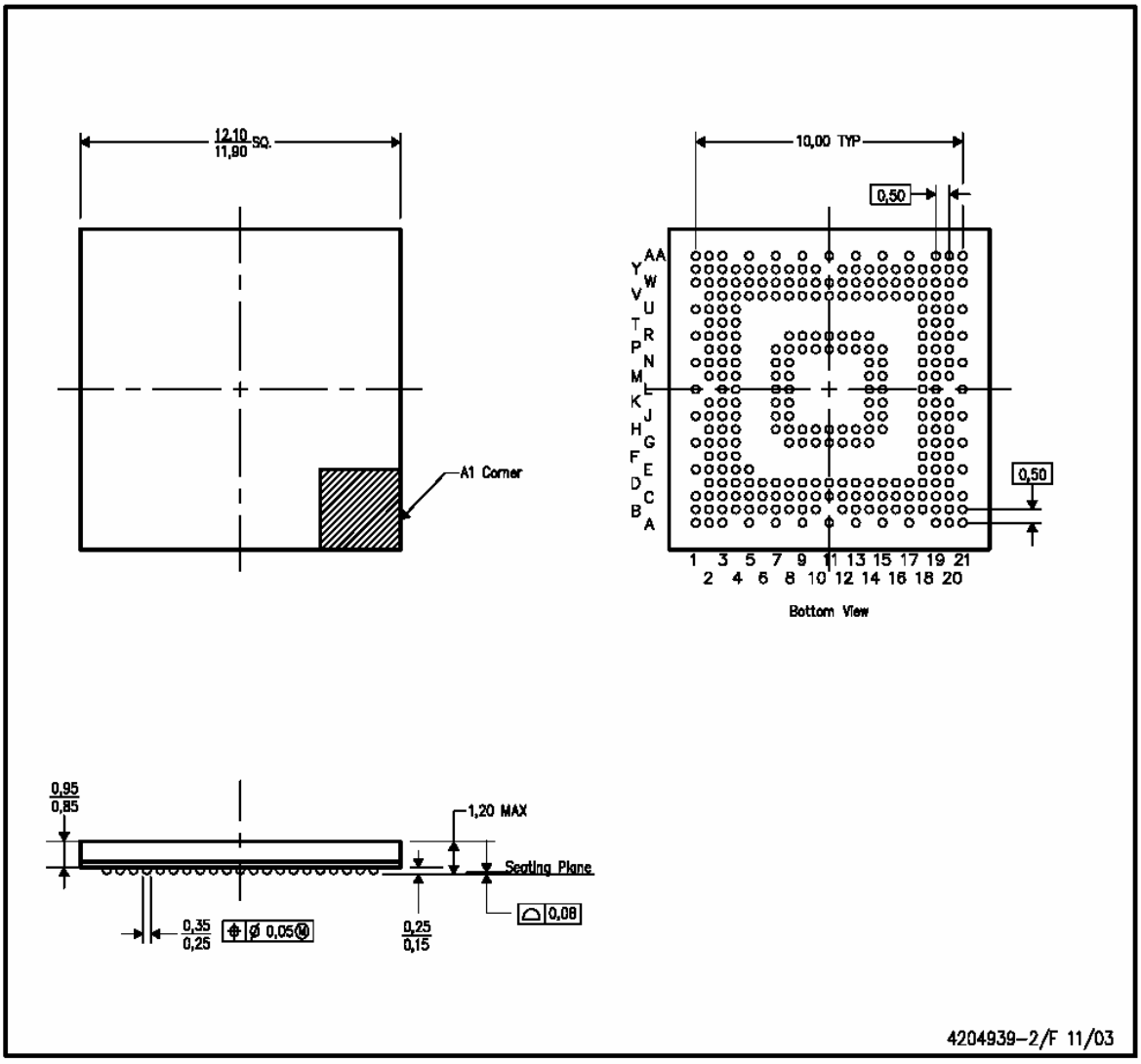


B.2 Mechanical Data

OMAP850 package is 289-pin S-PBGA (GVL289).

Package code	**289-GVL μ*BGA**
Signal balls	**288**
Power balls	**0**
Leadframe	
Package size:	12 X 12 mm
Ball pitch	0.5 mm

GVL (S–PBGA–N289) **PLASTIC BALL GRID ARRAY**



NOTES: A. All linear dimensions are in millimeters.
B. This drawing is subject to change without notice.

Appendix C

OMAP730–OMAP850 Cross-Reference

This appendix contains a copy of the OMAP730 TRM table of contents (TOC). Shaded text indicates those sections of the OMAP730TRM that have been modified to suit the particular features of the OMAP850 devices and that are part of the present document.

Unshaded sections apply to both devices—OMAP730/750. See the OMAP730TRM (literature number SWPU063B) for complete information on these sections.

Topic **Page**

C.1 TOC Cross-Reference C-2

C.1 TOC Cross-Reference

This appendix contains the TOC from the OMAP730 TRM. Shaded sections are modified and contained in the present book (OMAP850 TRM). Unshaded sections apply to OMAP730 and OMAP850 and can be found in the OMAP730 TRM.

Contents

1 Introduction to the OMAP730 System
- 1.1 Device Description
  - 1.1.1 OMAP730 Architecture
- 1.2 Features
  - 1.2.1 GSM-MPU Module
  - 1.2.2 DSP Subchip
  - 1.2.3 MPU Module
  - 1.2.4 Shared Module
- 1.3 Architecture
- 1.4 Memory Maps
  - 1.4.1 Memory
  - 1.4.2 GSM-S MPU Memory Map
  - 1.4.3 GSM-S DSP Memory Space
- 1.5 Security
  - 1.5.1 Main Characteristics
  - 1.5.2 Security Architecture
  - 1.5.3 Hardware Security
  - 1.5.4 Boot ROM and Security
  - 1.5.5 eFuses
  - 1.5.6 SHA1/MD5 Accelerator
  - 1.5.7 DES/3DES
  - 1.5.8 Random Number Generator (RNG)
  - 1.5.9 Secure Watchdog Timer

2 MPU Subsystem
- 2.1 OMAP730 Platform Description
- 2.2 Boot Sequences for OMAP730 ES1.0 Device Revision
  - 2.2.1 ROM Code Overview
  - 2.2.2 ROM Code Description
- 2.3 Boot Sequences for OMAP730 Revision ES1.1 (or Higher)
  - 2.3.1 Boot ROM Execution
  - 2.3.2 ROM Operating Modes
  - 2.3.3 Flashing
  - 2.3.4 Booting
  - 2.3.5 Secure Environment at Boot Time
  - 2.3.6 USB Boot Configurations

2.4 TIPB Bridge
2.4.1 Functionality
2.4.2 Registers
2.5 Memory Interface Traffic Controller
2.5.1 Description
2.5.2 OCP-T1/OCP-T2 Description
2.5.3 EMIFS Programming
2.5.4 EMIFF Programming
2.5.5 OCP-I Programming
2.5.6 Traffic Controller Registers
2.5.7 EMIFS Registers
2.5.8 EMIFF Registers
2.5.9 OCPI Registers
2.5.10 Priority Algorithms
2.6 Operating System Timers
2.6.1 Functionality
2.6.2 Timer Interrupts
2.6.3 Timer Programming
2.6.4 Registers
2.7 Watchdog Timers
2.7.1 Functionality
2.7.2 Registers
2.9 LED Pulse Generator
2.9.1 Features
2.9.2 LPG Design
2.9.3 LPG Power Management
2.9.4 LPG Registers
2.10 MPU Serial Port Interface
2.10.1 SPI Registers
2.10.2 Protocol
2.10.3 Transmission Mode Chronograms
2.10.4 SPI Inputs/Outputs
2.10.5 Timing Characterization
2.10.6 MPU TIPB Bus
2.10.7 Serial Interface
2.11 MPU General-Purpose Input/Output
2.11.1 GPIO Registers
2.11.2 Interrupt Logic Block
2.11.3 External GPIO Expansion
2.11.4 Extended GPIO Register
2.12 MPUIO
2.12.1 MPU I/O Interrupts
2.12.2 MPU I/O Clocks and Reset
2.12.3 Keyboard Interface
2.12.4 MPUIO Interface
2.12.5 MPUIO Interrupt Reset
2.12.6 MPUIO Interrupt Masking
2.12.7 Event Capture Module
2.12.8 MPU I/O Registers
2.13 MicroWire Interface (μWire)

2.13.1 MicroWire Registers
2.13.2 Protocol Description
2.13.3 Example of Protocol Using a Serial EEPROM (XL93LC66)
2.13.4 Example of Protocol Using an LCD Controller (COP472-3)
2.13.5 Example of Protocol Using Autotransmit Mode
2.13.6 Example of Autotransmit Mode With DMA Support
2.13.7 Limitations of μWIRE Interface
2.14 HDQ and 1-Wire Protocols
2.14.1 Functional Description
2.14.2 Power-Down Mode
2.14.3 HDQ and 1-Wire Battery Monitoring Serial Interface
2.14.4 Software Interface
2.15 Pulse-Width Tone Modulator
2.15.1 Overview
2.15.2 PWT Features
2.15.3 PWT Registers
2.15.4 PWT Programming
2.16 Pseudonoise Pulse-Width Light Modulator
2.16.1 PWL Functional Description
2.16.2 PWL Registers
2.17 32-kHz Timer
2.17.1 Operating System Scalable Clock-Tick Interrupt Function
2.17.2 32-kHz Timer Registers
2.18 LCD Low-Power Controller
2.18.1 General Description
2.18.2 Block Diagram
2.18.3 LLPC Functional Description
2.18.4 LLPC Registers
2.18.5 LLPC Registers Content
2.18.6 Required Timing
2.18.7 Interruptions and Requests
2.18.8 Software Procedures
2.19 DSP Memory Management Unit (DSP MMU0
2.20 Address Translation
2.20.1 Translation Process
2.20.2 Page Table Format
2.20.3 Coarse Page Tables
2.20.4 Fine Page Tables
2.21 Functionality
2.21.1 Translation Summary
2.21.2 Lock Mechanism and the Current_Victim Counter
2.21.3 Fault Handling
12.21.4 Initializing Locked TLB Entries
2.21.5 Table Walking Logic
2.21.6 Boot
2.22 DSP MMU Registers
2.23 Additional MPU Components

3 GSM Subsystem
3.1 Introduction
3.1.1 Features

3.2 GSM-MPU Core
3.3 DSP Subchip
3.4 CLKM
3.4.1 Functional Description
3.4.2 CLKM Registers
3.5 Real Time Clock (RTC
3.5.1 RTC REGISTERS
3.6 Configuration Registers
3.7 GSM-MPU Peripherals
3.7.1 Memory Interface
3.7.2 Internal Static RAM
3.7.3 Internal Boot Memory
3.7.4 Die ID Cell
3.7.5 Interrupt Handler (INTH)
3.7.6 General-Purpose I/O (GPIO)
3.7.7 Microwire Interfaces (mWIRE)
3.7.8 Timers
3.7.9 Universal Asynchronous Receiver/Transmitter 16C750 (UART Modem)
3.7.10 Subscriber Identity Module Interface (SIM) Interface
3.7.11 Serial Port Interface (SPI)
3.7.12 Time Processing Unit (TPU)
3.7.13 Time Serial Port (TSP)
3.7.14 Direct Memory Access (DMA) Controller
3.7.15 Clock Management (CLKM)
3.7.16 Light Pulse Generator (LPG)
3.7.17 I2C Master Serial Interface (I2C)
3.7.18 Memory Protection Unit (MPU)
3.7.19 Debug Unit (DU)
3.7.20 GPRS Encryption Algorithm (GEA1-2)
3.7.21 Internal RAM Write Buffer (WRB)
3.7.22 Real Time Clock (RTC)
3.7.23 Ultralow-Power-Down Controller
3.8 Peripherals Definition
3.8.1 GPRS Encryption Algorithm (GEA 1 & 2)
3.8.2 GEA Registers
3.8.3 Downlink Configuration Registers: CONF_DL_REG(1:5)
3.8.4 Ciphering Key Registers: KC_REG(1:4)
3.8.5 FCS Uplink Registers: FCS_UL_REG(1:2)
3.8.6 FCS Downlink Registers: FCS_DL_REG(1:2)
3.8.7 Data Register
3.8.8 Frame Bit Order
3.8.9 Frame Splitting
3.8.10 GEA Programming Schedule
3.8.11 DSP Access From/To MPUIC
3.9 DSP XIO to TIPB Registers—XIO:F800
3.10 MPUI Register
3.11 MPUIC Control Register—GSM-MPU Reads from MPUIC
3.12 DMA Mapping
3.13 DMA Registers
3.13.1 GSM-MPU Registers

3.14 Radio Interface Registers
3.14.1 Shift Data Registers (XSR and RSR)
3.15 Cipher Registers
3.16 MCSI Registers
3.16.1 Control Registers
3.16.2 Data Registers
3.17 DSP Interrupts
3.17.1 Internal Registers
3.18 GSM-MPU Interrupts—FFFF:FA00
3.18.1 Interrupt Sequence
3.18.2 GSM-MPU Interrupt Registers
3.18.3 GSM-MPU To TIPB Registers
3.18.4 DPLL REGISTER
3.20.5 DPLL Operation
3.20.6 Lock Times
3.20.7 Control Register Access
3.20.8 Timer Registers
3.20.9Watchdog Timer Registers
3.20.10SPI Registers
3.20.11 µWIRE Registers
3.20.13MPU I/O Registers
3.19 SIM Registers
3.20 Time Serial Port (TSP) Registers
3.20.1 Parallel Bit Interface
3.20.2 TSP Receive and Transmit Registers
3.20.3 TPU Sequencer Internal Address Registers Mapping
3.21 TPU Registers
3.21.1 TPU RAM Memory Mapping
3.22 TPU Sequencer
3.22.1 Functional Description
3.22.2 Instruction Execution Flow
3.22.3 Microinstruction Set Definition
3.22.4 Structure of the Microinstruction
3.22.5 TPU Instruction Set
3.23 ULPD Registers
3.24 LPG Registers
3.24.1 Design Constraint
3.25 UART 16C750 Registers
3.25.1 Line Status Registers (LSR)
3.25.2 Interrupt Enable Register (IER)
3.25.3 Interrupt Identification Register (IIR)
3.26 I2C Registers
3.26.1 I2C Features
3.27 DSP Peripherals
3.27.1 Radio Interface (RIF)
3.27.2 Multichannel Serial Interface (MCSI)
3.27.3 Ciphering Processor (CRYPT)
3.27.4 Universal Asynchronous Receiver/Transmitter (16C750)
3.27.5 Direct Memory Access Controller (DMA)
3.27.6 Interrupt Handler (INTH)

3.27.7 DSP Program/Data Memory Extension
3.28 GSM-S Memory Mapping
3.28.1 Memory Interface Mapping
3.28.2 External Flash/ROM Image
3.28.3 TIPB Peripherals Mapping
3.28.4 GSM-S DSP Memory Space
3.28.5 MPUIF
3.28.6 XIO Memory Mapping
3.28.7 XIO−TIPB
3.29 GSM-S Interrupts Mapping
3.29.1 MPU Interrupts
3.30 GSM-S DMA Mapping
3.30.1 GSM-S DMA Requests
3.31 GSM Memory Protection
3.31.1 System Architecture Overview
3.31.2 Functional Overview
3.31.3 GSM Memory Space Mapping and Protection Definition
3.31.4 GSM/DSP MMU Restrictions
3.31.5 Registers Implementation
3.31.6 Violation Handler
3.31.7 Reset Mode/Debug Mode State Machine
3.31.8 Security Guidelines
3.31.9 Using the GSM Protect to Define a Read-Only SDRAM Platform Zone

4 Traffic Controller Interface
4.1 Traffic Controller Interface Module
4.1.1 Common Features of the Four Memory Zones
4.2 TCIF Module Functionality
4.2.1 Module Behavior
4.2.2 Module Functionality
4.2.3 Module Registers
4.3 Using TCIF Software
4.3.1 Memory Management
4.3.2 Buffer and Cache
4.4 TCIF Module Architecture
4.4.2 Module Interface Description
4.5 Intersystem Communication Register
4.5.1 Module Overview
4.5.2 Module Behavior
4.5.3 Module Specification
4.5.4 Top-Level Block Diagram
4.5.5 I/O Timing Diagram
4.6 OMAP730 Configuration Module
4.6.1 Configuration Register Capabilities
4.6.2 Configuring Pin Multiplexing and Pullups/Pulldowns
4.6.3 Configuring the USB
4.6.4 Programming Security Registers
4.6.5 Configuring Supply Voltage
4.7 OMAP730 Configuration Registers
4.8 Inputs/Outputs
4.9 MPU/GSM: Shared Port

5 Clock Generation and Reset Module
- 5.1 Module Description
- 5.2 Clock Generation
  - 5.2.1 Clock Generation Modes
  - 5.2.2 DPLL
  - 5.2.3 MPU Clock Domain
  - 5.2.4 Modem Connection Clock Domain
  - 5.2.5 Traffic Controller Clock Domain
- 5.3 Power-Saving Modes and Wake-Up Control
  - 5.3.1 MPU Idle Control
  - 5.3.2 Traffic Controller, System DMA Controller, and MPU TIPB Bridges Idle Control
  - 5.3.3 External Device Power Control
  - 5.3.4 DPLL Idle Control
  - 5.3.5 Chip Idle Mode, Deep Sleep Mode, and Wake-up Control
- 5.4 System Reset
- 5.5 Registers
  - 5.5.1 MPU Registers
  - 5.5.2 DPLL Registers

6 MPU-S Interrupt Handler
- 6.1 Description
  - 6.1.1 Interrupt Control and Configuration
  - 6.1.2 Software Interrupt
- 6.2 Interrupt Sequence
- 6.3 Interrupt Handler Software
  - 6.3.1 Edge-Triggered Interrupts
  - 6.3.2 Level-Sensitive Interrupts
- 6.4 Registers
  - 6.4.1 Interrupt Registers
  - 6.4.2 Interrupt Level 2 Registers

7 MPU-S Direct-Memory Access
- 7.1 General Overview
- 7.2 Functional Description
  - 7.2.1 Logical Channel Types
  - 7.2.2 OMAP730 System DMA Instances
  - 7.2.3 Synchronized Channel
  - 7.2.4 Physical Ports
  - 7.2.5 Port Channel Scheduling
  - 7.2.6 Logical Channel Scheduling
  - 7.2.7 Logical Channel Interleaving For Synchronized Transfers
  - 7.2.8 Linking Logical Channels
  - 7.2.9 Logical Channel Preempting
  - 7.2.10 Addressing Modes
  - 7.2.11 Data Packing and Bursting
  - 7.2.12 Interrupt Generation
  - 7.2.13 DMA Request Input Protection
  - 7.2.14 DMA Idle Modes
  - 7.2.15 DMA Debug State
  - 7.2.16 Other Logical Channel Features
  - 7.2.17 Compatibility with OMAP 3.0 and 3.1

7.3 LCD Channel
7.3.1 Display Logical Channel
7.3.2 LCD Channel Addressing Modes
7.3.3 DMA LCD Channel Sharing Feature
7.3.4 DMA LCD Channel Rotation
7.3.5 DMA LCD Channel Autoinitialization Feature
7.3.6 DMA_LCD_Disable/Bus Error Feature
7.3.7 LCD Channel Usage Restrictions
7.3.8 LCD Channel OMAP 3.0/3.1 Compatible Mode Programming
7.4 System DMA Registers
7.4.1 Summary DMA Global Registers
7.4.2 Logical Channel Registers
7.4.3 LCD Channel Dedicated Registers

8 LCD Controller
8.1 LCD Controller Environment
8.2 LCD Controller Operation
8.2.1 Frame Buffer
8.2.2 Control Blocks
8.2.3 Interrupts
8.2.4 LCD Subpanel Display Support
8.3 Registers
8.3.1 LCD Control Register (LCDCONTROL)
8.3.2 LCD Timing 0 Register (LcdTiming0)
8.3.3 LCD Timing 1 Register (LcdTiming1)
8.3.4 LCD Timing 2 Register (LcdTiming2)
8.3.5 LCD Controller Status Register (LcdStatus)
8.3.6 LCD Subpanel Display Register (LcdSubpanel)
8.3.7 Line Interrupt Register (LcdLineInt)
8.3.8 Display Status Register (LcdDisplayStatus)

9 UART Modem/IrDA
9.1 Signals and Block Diagram
9.2 Main Features
9.2.1 Modem Functions
9.2.2 IrDA Functions
9.3 UART Modem/IrDA Registers
9.4 Modes of Operation
9.4.1 Modem Mode
9.4.2 SIR Mode
9.4.3 MIR Mode
9.4.4 MIR Transmit Frame Format
9.4.5 FIR Mode
9.5 Functional Description
9.5.1 Trigger Levels
9.5.2 Interrupts
9.5.3 FIFO Interrupt Mode Operation
9.5.4 FIFO Polled Mode Operation
9.5.5 FIFO DMA Mode Operation
9.5.6 Sleep Mode
9.5.7 IrDA Modes

9.5.8 Idle Modes
9.5.9 Break and Time-Out Conditions
9.5.10 Programmable Baud Rate Generator
9.5.11 Hardware Flow Control
9.5.12 Software Flow Control
9.5.13 Autobauding Mode
9.5.14 Frame Closing
9.5.15 Store and Controlled Transmission (SCT)
9.5.16 Underrun During Transmission
9.5.17 Overrun During Receive
9.5.18 Status FIFO
9.6 UART Configuration Example
9.6.1 UART Software Reset
9.6.2 UART FIFO Configuration
9.6.3 Baud Rate Data and Stop Configuration

10 Universal Serial Bus
10.1 Overview
10.2 USB Host Controller
10.2.1 USB Open Host Controller Interface Functionality
10.2.2 OMAP730 USB Host Controller Differences From OHCI Specification for USB
10.2.3 OMAP730 Implementation of OHCI Specification for USB
10.2.4 USB Host Controller Registers
10.2.5 USB Host Controller Reserved Registers and Reserved Bit Fields
10.2.6 USB Host Controller Registers, USB Reset, and USB Clocking
10.2.7 OHCI Interrupts
10.2.8 USB Host Controller Access to System Memory
10.2.9 Physical Addressing
10.2.10 Cache Coherency in OHCI Data Structures and Data Buffers
10.2.11 OCPI Bus Addressing and OHCI Data Structure Pointers
10.2.12 NULL Pointers
10.2.13 OMAP730 OCPI Bus and the USB Host Controller
10.2.14 OCPI Registers
10.2.15 USB Host Controller Clock Control
10.2.16 USB Host Controller Hardware Reset
10.2.17 USB Host Controller OHCI Reset
10.2.18 USB Host Controller Power Management
10.2.19 OCPI Clocking
10.3 USB Device Controller
10.3.1 USB Device Controller Registers
10.3.2 USB Device Transactions
10.3.3 Non-Isochronous, Non-Setup OUT (USB HOST -> MPU) Transactions
10.3.4 Non-Isochronous IN (MPU->USB HOST) Transactions
10.3.5 Isochronous OUT (USB HOST-> MPU) Transactions
10.3.6 Isochronous IN (MPU->USB HOST) Transactions
10.3.7 Control Transfers on Endpoint 0
10.3.8 USB Device Initialization
10.3.9 Preparing for Transfers
10.3.10 USB Device Interrupt Service Routine (ISR) Flowcharts
10.3.11 Important Note on USB Device Interrupts
10.3.12 Parsing General USB Device Interrupt

10.3.13 Setup Interrupt Handler
10.3.14 Endpoint 0 RX Interrupt Handler
10.3.15 Endpoint 0 TX Interrupt Handler
10.3.16 Device States Changed Handler
10.3.17 Device States Attached/Unattached Handler
10.3.18 Device State Configuration Changed Handler
10.3.19 Device State Address Changed Handler
10.3.20 USB Device Reset Interrupt Handler
10.3.21 Suspend/Resume Interrupt Handler
10.3.22 Parsing Non-ISO Endpoint-Specific Interrupt
10.3.23 Non-ISO, Non-Control OUT Endpoint Receive Interrupt Handler
10.3.24 Non-ISO, Non-Control IN Endpoint Transmit Interrupt Handler
10.3.25 SOF Interrupt Handler
10.3.26 Summary of USB Device Controller Interrupts
10.3.27 DMA Operation
10.3.28 Power Management
10.4 USB OTG Controller
10.4.1 OTG Controller Features
10.4.2 OTG Controller Registers
10.4.3 Pin Multiplexing
10.4.4 Selecting and Configuring USB Connectivity
10.4.5 Transceiver Signaling Types
10.4.6 USB OTG External Connectivity
10.4.7 Host Controller Connectivity With USB Transceivers
10.4.8 USB Function Controller Connectivity With USB Transceivers
10.4.9 Onboard Transceiverless Connection Using OMAP730 Transceiverless Link
10.4.10 Conflicts Between USB Signal Multiplexing and Top-Level Multiplexing
10.4.11 OMAP730 USB Hardware Considerations

11 Multimedia Card (MMC/SD/SDIO) Interface
11.1 MMC Overview
11.1.1 MMC/SD/SDIO Host Controller Features
11.1.2 MMC/SD Host Controller Signal Pads
11.1.3 MMC.CLK, SPI.CLK Signals ac Characteristics
11.1.4 MMC/SD/SDIO Modes—Interface Signal ac Characteristics
11.1.5 SPI Mode—Interface Signal ac Characteristics
11.2 MMC Registers
11.3 MMC Command Flow
11.3.1 Basic Operations
11.3.2 System Test Mode
11.3.3 SPI Mode
11.4 DMA Operations
11.4.1 MMC DMA Receive Mode
11.4.2 MMC DMA Transmit Mode
11.4.3 SDIO Suspend/Resume
11.4.4 Programming Model Incompatibility

12 McBSPs
12.1 Introduction to McBSPs
12.1.1 Key Features of the McBSPs
12.1.2 McBSP Generic Block Diagram

12.1.3 McBSP Pins
12.1.4 McBSP Register Addresses
12.2 McBSP Operation
12.2.1 Data Transfer Process of McBSPs
12.2.2 Companding (Compressing and Expanding) Data
12.2.3 Clocking and Framing Data
12.2.4 Frame Phases
12.2.5 McBSP Reception
12.2.6 McBSP Transmission
12.3 McBSP Sample Rate Generator
12.3.1 Clock Generation in the Sample Rate Generator
12.3.2 Frame-Synchronization Generation in the Sample Rate Generator
12.3.3 Synchronizing Sample Rate Generator Outputs to an External Clock
12.3.4 Reset and Initialization Procedure for the Sample Rate Generator
12.3.5 Sample Rate Generator Clocking Examples
12.4 McBSP Exception/Error Conditions
12.4.1 Overrun in the Receiver
12.4.2 Unexpected Receive Frame-Synchronization Pulse
12.4.3 Overwrite in the Transmitter
12.4.4 Underflow in the Transmitter
12.4.5 Unexpected Transmit Frame-Synchronization Pulse
12.5 Multichannel Selection Modes
12.5.1 Channels, Blocks, and Partitions
12.5.2 Multichannel Selection
12.5.3 Configuring a Frame for Multichannel Selection
12.5.4 Using Two Partitions
12.5.5 Using Eight Partitions
12.5.6 Receive Multichannel Selection Mode
12.5.7 Transmit Multichannel Selection Modes
12.5.8 Using Interrupts Between Block Transfers
12.6 SPI Operation Using the Clock Stop Mode
12.6.1 SPI Protocol
12.6.2 Clock Stop Mode
12.6.3 Bits Used to Enable and Configure the Clock Stop Mode
12.6.4 Clock Stop Mode Timing Diagrams
12.6.5 Procedure for Configuring a McBSP for SPI Operation
12.6.6 McBSP as the SPI Master
12.6.7 McBSP as an SPI Slave
12.7 Receiver Configuration
12.7.1 Programming the McBSP Registers for the Desired Receiver Operation
12.7.2 Resetting and Enabling the Receiver
12.7.3 Set the Receiver Pins to Operate as McBSP Pins
12.7.4 Enable/Disable the Digital Loopback Mode
12.7.5 Enable/Disable the Clock Stop Mode
12.7.6 Enable/Disable the Receive Multichannel Selection Mode
12.7.7 Choose One or Two Phases for the Receive Frame
12.7.8 Set the Receive Word Length(s)
12.7.9 Set the Receive Frame Length
12.7.10 Enable/Disable the Receive Frame-Synchronization Ignore Function
12.7.11 Set the Receive Companding Mode

12.7.12 Set the Receive Data Delay
12.7.13 Set the Receive Sign-Extension and Justification Mode
12.7.14 Set the Receive Interrupt Mode
12.7.15 Set the Receive Frame-Synchronization Mode
12.7.16 Set the Receive Frame-Synchronization Polarity
12.7.17 Set the SRG Frame-Synchronization Period and Pulse Width
12.7.18 Set the Receive Clock Mode
12.7.19 Set the Receive Clock Polarity
12.7.20 Set the SRG Clock Divide-Down Value
12.7.21 Set the SRG Clock Synchronization Mode
12.7.22 Set the SRG Clock Mode (Choose an Input Clock)
12.7.23 Set the SRG Input Clock Polarity
12.8 Transmitter Configuration
12.8.1 Programming the McBSP Registers for the Desired Transmitter Operation
12.8.2 Resetting and Enabling the Transmitter
12.8.3 Set the Transmitter Pins to Operate as McBSP Pins
12.8.4 Enable/Disable the Digital Loopback Mode
12.8.5 Enable/Disable the Clock Stop Mode
12.8.6 Enable/Disable Transmit Multichannel Selection
12.8.7 Choose One or Two Phases for the Transmit Frame
12.8.8 Set the Transmit Word Length(s)
12.8.9 Set the Transmit Frame Length
12.8.10 Enable/Disable the Transmit Frame-Synchronization Ignore Function
12.8.11 Set the Transmit Companding Mode
12.8.12 Set the Transmit Data Delay
12.8.13 Set the Transmit DXENA Mode
12.8.14 Set the Transmit Interrupt Mode
12.8.15 Set the Transmit Frame-Synchronization Mode
12.8.16 Set the Transmit Frame-Synchronization Polarity
12.8.17 Set the SRG Frame-Synchronization Period and Pulse Width
12.8.18 Set the Transmit Clock Mode
12.8.19 Set the Transmit Clock Polarity
12.8.20 Set the SRG Clock Divide-Down Value
12.8.21 Set the SRG Clock Synchronization Mode
12.8.22 Set the SRG Clock Mode (Choose an Input Clock)
12.8.23 Set the SRG Input Clock Polarity
12.9 General-Purpose I/O on the McBSP Pins
12.10 Emulation, Power, and Reset Considerations
12.10.1 McBSP Emulation Mode
12.10.2 Reducing Power Consumed by McBSPs
12.10.3 Resetting and Initializing McBSPs
12.11 Data Packing Examples
12.11.1 Data Packing Using Frame Length and Word Length
12.11.2 Data Packing Using Word Length and the Frame-Synchronization Ignore Function
12.12 McBSP on the OMAP730 Device—Applications
12.12.1 Communication McBSP Interface
12.12.2 I2S Audio Codec McBSP Interface
12.13 McBSP Registers
12.13.1 Data Receive Registers (DRR2 and DRR1)

12.13.2 Data Transmit Registers (DXR2 and DXR1)
12.13.3 Serial Port Control Registers (SPCR1 and SPCR2)
12.13.4 Receive Control Registers (RCR1 and RCR2)
12.13.5 Transmit Control Registers (XCR1 and XCR2)
12.13.6 Sample Rate Generator Registers (SRGR1 and SRGR2)
12.13.7 Multichannel Control Registers (MCR1 and MCR2)
12.13.8 Pin Control Register (PCR)
12.13.9 Receive Channel Enable Registers (RCERA, RCERB, RCERC, RCERD, RCERE, RCERF, RCERG, RCERH)
12.13.10 Transmit Channel Enable Registers (XCERA, XCERB, XCERC, XCERD, XCERE, XCERF, XCERG, XCERH)
12.14 McBSP Register Worksheet
12.14.1 General Control Registers
12.14.2 Multichannel Selection Control Registers

13 NAND Flash
13.1 Hardware NAND Flash Controller
13.1.1 Read Operation
13.1.2 Write Operation
13.1.3 Multiplane Page Program
13.1.4 Erase Operation
13.1.5 Multiplane Block Erase Operation
13.1.6 Multiplane Copy-Back Program Operation
13.1.7 Read Status and Read Multiplane Status Operations
13.1.8 Reset Operation
13.1.9 Read ID Operation
13.1.10 Error Code Correction
13.1.11 Invalid Block Management
13.1.12 FIFO (Prefetch and Postwrite)
13.1.13 Prefetch
13.1.14 Postwrite
13.1.15 DMA Support
13.1.16 Host Mode
13.1.17 FIFO mode
13.1.18 NAND Flash Memory Core Support
13.1.19 NAND Flash Registers
13.2 Software NAND Flash Controller
13.2.1 EMIFS Interface With NAND CE Care Flash Device Option
13.2.2 Write Data Sequence Example
13.2.3 Read Data Sequence Example
13.2.4 EMIFS Interface With NAND CE Don't Care Flash Device Option
13.2.5 NAND Flash Controller Peripheral/NOR Flash Add-On Option

14 MPU-S I2C Serial Interface
14.1 I^2C Master/Slave
14.1.1 Overview
14.1.2 Functional Overview
14.1.3 I^2C Controller Features
14.1.4 I^2C Master/Slave Controller Signal Pads
14.1.5 I^2C Reset
14.1.6 I^2C Bit Transfer
14.1.7 Data Validity

14.1.8 START and STOP Conditions
14.2 I2C Operation
14.2.1 Serial Data Formats
14.2.2 I2C Interrupts
14.2.3 DMA Events
14.3 Register Map
14.4 Programming Guidelines
14.4.1 Main Program
14.4.2 Interrupt Subroutines
14.4.3 Flow Diagrams

15 Enhanced Audio Controller (EAC)
15.1 Introduction
15.2 General Description
15.3 Features
15.4 EAC and EAC-2 Comparison
15.4.1 EAC-2 New Features Description
15.4.2 EAC-2 Programming Model
15.5 Architecture
15.5.1 Functional Block Diagram
15.5.2 Memory-Mapped Registers
15.6 Clock Manager
15.6.1 Description
15.6.2 Configuration
15.6.3 External Audio Oscillator Support for Codec Master Clock
15.7 Codec Port Interface
15.7.1 Description
15.7.2 Configuration
15.7.3 Codec Port Programming Limits
15.7.4 AC'97 (Audio Codec '97) Mode
15.7.5 Inter-IC Sound (I2S) Mode
15.7.6 Pulse Code Modulation (PCM) Mode
15.8 Modem Port Interface
15.8.1 Description
15.8.2 Configuration
15.8.3 Main Channel Protocol Chronograms
15.8.4 Auxiliary Channel Protocol Chronograms
15.9 Bluetooth Port Interface
15.9.1 Description
15.9.2 Configuration
15.10 Sample-Rate Converter
15.10.1 Description
15.10.2 Configuration
15.10.3 Group Delay Inserted by SRCs
15.11 DMA Channels
15.11.1 Description
15.11.2 Configuration
15.12 μ-Law/Law Companding
15.12.1 Description
15.12.2 Configuration
15.13 Sidetone

15.13.1 Description
15.13.2 Configuration
15.14 Mixer
15.14.1 Description
15.14.2 Configuration
15.15 DMA Volume Control
15.15.1 Description
15.15.2 Configuration
15.16 Peak Detectors
15.16.1 Description
15.17 Application: Normal Phone Call
15.18 Application: Normal Phone Call With Record
15.19 Application: Normal Phone Call With Play
15.20 Application: Normal Phone Call With Music
15.21 Application: Communication With Headset
15.22 Application: Communication With Headset and Record
15.23 Application: Communication With Headset and Music
15.24 Application: Record a Message
15.25 Application: Listen to Music
15.26 Application: Record a Message From Headset
15.27 Application: Listening to Music With Headset
15.28 Application: Display Wave File Stored in Flash Memory
15.28.1 Description
15.28.2 EAC Configuration
15.28.3 DMA Controller Configuration
15.29 Memory-Mapped registers

16 Multichannel Serial Interface (MCSI)
16.1 Multichannel Serial Interface
16.1.1 Communication Protocol
16.1.2 MCSI Registers
16.2 MPU_MCSI
16.2.1 MPU_MCSI Pin Description
16.2.2 MPU_MCSI Interrupt Mapping
16.2.3 MPU_MCSI DMA Request Mapping
16.2.4 MCSI Addresses and Mapping

17 Power and Control Clock
17.1 General Description
17.1.1 Power and Clock Control Features
17.1.2 Functional Description
17.2 Functional Description
17.2.1 ULPD Subsystem
17.2.2 Power-Up Mechanism
17.2.3 Wake-Up Mechanism
17.2.4 GSP-MPU Wake-Up Interaction
17.2.5 MPU and DBB Sleep Sequence
17.2.6 Low-Dropout Voltage (LDO) Management
17.2.7 Timing Diagram
17.2.8 APLL 32-kHz to 1MHz Wrapper Overview
17.2.9 Clock Switching Conditions

17.2.10 APLL 96-MHz Input Clock Switch
17.2.11 NIRQ Generation
17.3 Peripherals Interface
17.3.1 PCC Peripheral Clock Constraints
17.4 PCC to ULPD Look-Up Tables
17.4.1 Registers
17.4.2 Software-Request
17.5 Generated PCC Clocks
17.6 Registers
17.7 Timing Diagrams
17.7.1 First Power-On Waveform
17.7.2 On-to-Off Waveform
17.7.3 Off-to-On Waveform
17.8 Embedded LDO Management
17.8.1 Requirements
17.8.2 LDO Management Scheme
17.8.3 Timing Diagrams
17.8.4 Using an External Supply for the PLL

18 VLYNQ Serial Communications Interface
18.1 Introduction
18.2 Operation
18.2.1 Overview
18.2.2 Write Operation
18.2.3 Read Operation
18.2.4 Initialization
18.2.5 Address Translation
18.2.6 Clocking
18.2.7 Interrupts
18.2.8 Flow Control
18.3 Packet Structure
18.3.1 Special Code Groups
18.3.2 Supported Ordered Sets
18.3.3 Packet Format
18.4 Registers
18.4.1 Revision Register (Base Address + 0x00)
18.4.2 Control Register (Base Address + 0x04)
18.4.3 Status Register (Base Address + 0x08)
18.4.4 Interrupt Status/Clear Register (Base Address + 0x10)
18.4.5 Interrupt Pending/Set Register (Base Address + 0x14)
18.4.6 Interrupt Pointer Register (Base Address + 0x18)
18.4.7 Tx Address Map Register (Base Address + 0x1c)
18.4.8 Rx Address Map Size 1 Register (Base Address + 0x20)
18.4.9 Rx Address Map Offset 1 Register (Base Address + 0x24)
18.4.10 Rx Address Map Size 2 Register (Base Address + 0x28)
18.4.11 Rx Address Map Offset 2 Register (Base Address + 0x2c)
18.4.12 Rx Address Map Size 3 Register (Base Address + 0x30)
18.4.13 Rx Address Map Offset 3 Register (Base Address + 0x34)
18.4.14 Rx Address Map Size 4 Register (Base Address + 0x38)
18.4.15 Rx Address Map Offset 4 Register (Base Address + 0x3c)
18.4.16 Chip Version Register (Base Address + 0x40)

18.4.17 Interrupt Vector 3−0 Register (Base Address + 0x60)
18.4.18 Interrupt Vector 7−4 Register (Base Address + 0x64)
18.4.19 Remote Configuration Registers (Base Address 0x80−0xfc)
18.5 Pins
18.6 Electrical Information
18.7 Application Examples
18.7.1 Integrated Access Device
18.8 General Description
18.8.1 VLYNQ2OCP Module Overview
18.8.2 Block Diagram
18.9 Submodule Functional Description
18.9.1 VLYNQ Module
18.9.2 OCP Wrapper
18.10 Signal Description
18.11 OCP-VBUS Interface Wrapper
18.11.1 OCP2VBUS (OCP Slave)
18.11.2 VBUS2OCP (OCP Master)
18.12 VLYNQ Submodule
18.12.1 VLYNQ Module Register Memory Map
18.12.2 VLYNQ Module Address Translation
18.12.3 VLYNQ ASIC Memories
18.12.4 VLYNQ Initialization
18.13 VLYNQ2OCP Configuration
18.13.1 Clock Management
18.13.32Reset Management
18.14 VLYNQ2OCP Power-Down Mode
18.14.1 VLYNQ2OCP Implementation
18.14.2 IDLE Mode Implementation
18.14.3 VLYNQ Serial Clock Frequency Slow Down
18.15 VLYNQ2OCP Timing Diagram

19 Security Features
19.1 Security Features
19.1.1 Hardware Security
19.1.2 Secure Mode
19.1.3 Security State Machine
19.1.4 Secure Mode Options
19.1.5 ULPD
19.2 Security Control Register
19.3 Security eFuse
19.3.1 Security Keys
19.3.2 Electrical Fuse
19.4 Boot ROM
19.4.1 Access Control Management
19.5 Secure RAM
19.5.1 Secure SRAM
19.5.2 Access Management
19.6 Secure Watchdog
19.7 Secure Hardware Accelerators
19.7.1 DES3DES Module
19.7.2 Random Number Generator

19.7.3 SH1/MD5 Module
19.8 Debug Support Versus Security
19.8.1 Production eFuse
19.8.2 STI
19.8.3 Emulator Device
19.8.4 Normal Device

20 Smart Card Controller
20.1 SmartCard Controller (SMC)
20.1.1 Characteristics of Smart Cards
20.1.2 Smart Card Interface Features
20.1.3 Functional Description
20.1.4 Functional Operating Modes
20.1.5 Warm Reset Procedure
20.1.6 Sleep Mode (Clock Stop Mode)
20.1.7 Deactivation Sequence
20.2 SMC Protocol
20.2.1 Bit Duration
20.2.2 Timer Descriptions
20.3 SMC Architecture
20.3.1 TIPB Interface Submodule
20.3.2 Clock Submodule
20.3.3 Sequencer Submodule
20.3.4 Transmit/receive Submodule
20.3.5 Interrupt Controller
20.3.6 Registers Block
20.3.7 Card Presence Detection Submodule (Card Detect Debouncing)
20.4 SMC Register Description
20.5 SMC Interface
20.5.1 General Description
20.5.2 USIM Interface
20.6 Baud Rates Supported by SMC
20.6.1 General Description
20.6.2 Supported (F,D) Pairs
20.6.3 FSCLK = 13/1 MHz
20.6.4 FSCLK = 13/2 MHz
20.6.5 FSCLK = 13/4 MHz
20.6.6 FSCLK = 13/8 MHz

21 Dual-Mode Timer
21.1 Dual-Mode Timer
21.1.1 Description
21.1.2 Mode Functionality
21.1.3 Capture Mode Functionality
21.1.4 Compare Mode Functionality
21.1.5 Prescaler Functionality
21.1.6 Pulse-Width Modulation
21.1.7 Timer Interrupt Control
21.1.8 Sleep Mode Request and Acknowledge
21.1.9 Timer Counting Rate
21.1.10 Dual-Mode Timer Under Emulation

21.1.11 Accessing Registers
21.1.12 Programming Timer Registers
21.1.13 Reading Timer Registers
21.1.14 Writing Timer Registers
21.1.15 Dual-Mode Timer Registers
21.1.16 Implementation

22 Camera Interface
22.1 Camera Parallel Interface
22.1.1 Functional Architecture
22.1.2 Clock Switching Procedures
22.2 Camera Interface Bandwidth

23 Real-Time Clock
23.1 Real-Time Clock
23.1.1 Split Power Overview
23.1.2 Internal Level Shifters
23.1.3 Split Power Module
23.1.4 Output Control
23.1.5 On-Chip Reset Generation
23.1.6 Using Split Power
23.1.7 Interrupt Management
23.1.8 Oscillator Drift Compensation
23.1.9 Split Power Compatibility
23.1.10 RTC Registers

24 Memory Mapping
24.1 MPU-S Memory Mapping
24.1.1 MPU Memory Space
24.2 GSM-MPU Memory Mapping
24.2.1 GSM-MPU Memory Mapping
24.3 External Flash ROM Image
24.4 GSM-S DSP Memory Space
24.4.1 MPU Shared Memory
24.4.2 XIO Memory Mapping
24.4.2 XIO-TIPB

25 Interrupt Mapping
25.1 MPU-S Interrupt Mapping
25.2 GSM-S Interrupt Mapping
25.2.1 GSM-MPU Interrupt Mapping
25.2.2 DSP Interrupts Mapping

26 DMA Requests
26.1 MPU-S DMA Requests
26.2 GSM-S DMA Requests

A MPU-S Registers
A.1 MPUIO Registers
A.2 Camera_IF Registers
A.4 Clock Generation and Reset (CLKRST) Registers
A.5 DES3DES Registers

A.6 DMA Controller Registers
A.7 DPLL1 Registers
A.8 Dual-Mode Timer Registers
A.9 EAC Registers
A.10 EMIFF Registers
A.11 EMIFS Registers
A.12 Endianism Registers
A.13 GPIO Registers
A.14 GSM_Protect Registers
A.15 HDQ_1Wire Registers
A.16 I2C Registers
A.17 ICR Registers
A.18 MPU INTH Registers
A.19 LCDC Registers
A.20 Level2_INTH REGISTERS
A.21 LLPC Registers
A.22 McBSP1/McBSP2 Registers
A.23 MMC_SDIO Registers
A.24 LPG Registers
A.25 McBSP1 and McBSP2 Registers
A.26 MMC_SDIO Registers
A.27 NAND_Flash Registers
A.28 OCPI Registers (FFFE:C320)
A.29 OCPT Registers
A.30 PCC_ULPD Registers
A.31 PWL Registers
A.32 PWT Registers
A.33 RNG Registers
A.34 RTC Registers
A.35 Swatchdog Registers
A.36 SH1/MD5 Registers
A.37 SPI_100K_1 and SPI_100K_2 Registers
A.38 SYREN_SPI Registers
A.39 TCIF Registers
A.40 TESTBLOCK Registers
A.41 TIMER32K Registers
A.42 ARMT1, ARMT2, and ARMT3 Registers
A.43 ARMRHEA_PRIV Registers
A.44 ARMRHEA_PUB Registers
A.45 UART_MODEM and UART_MODEM_IRDA Registers
A.46 USB_OTG_CTRL Registers
A.47 USB_OTG_HHC Registers
A.48 USB_OTG_W2FC Registers
A.49 USIM Registers
A.50 UWIRE Registers
A.51 VLYNQ2OCP Registers
A.52 ARMWDG Registers

B GSM Subsystem Registers

B.1 API Registers – XIO:F900 – FFE0:0000
- B.1.1 DSP API Configuration Register Mapping
- B.1.2 APIC Control Register
- B.1.3 DSP Accesses From/to APIC

B.2 CLKM Registers
- B.2.1 MPU Clock Control Register (R/W)
- B.2.2 MPU Control Register (R/W)
- B.2.3 MPU Reset Control Register (R/W)
- B.2.4 Clock Division Factor Control Register (R/W)
- B.2.5 Free-Running Clock Control Register (R/W)

B.3 Cypher Registers – XIO:2800
- B.3.1 Cypher Register Mapping
- B.3.2 Cypher Control Register
- B.3.3 Interrupt Status Register
- B.3.4 Work Status Register (STATUS_work_REG) – XIO:2802
- B.3.5 KC Registers 1 to 4—XIO:2803 .. 2806
- B.3.6 Count Registers 1 and 2—XIO:2807.. 2808
- B.3.7 Decipher Data Registers 1 to 8 (DECI_REG_#) XIO 2809 .. 2810
- B.3.8 Encipher Data Register 1 to 8 (ENCI_REG_#) – XIO:2811 .. 2818

B.4 DMA Controller – FFFF:E800 – XIO:E800
- B.4.1 DMA Register Mapping
- B.4.2 Global Registers
- B.4.3 Regular Channel Registers

B.5 DSP Interrupts – XIO:FA00
- B.5.1 Internal Registers XIO:FA00 .. FA01

B.6 DSP XIO to TIPB– XIO:F800
- B.6.1 XIO-TIPB Register Mapping
- B.6.2 Transfer Rate Register (Read/Write) – XIO:F800
- B.6.3 Bridge Control Register (Read/Write) – XIO:F801
- B.6.4 NMI Register

B.7 RIF Registers – FFFF:7000 – XIO:0000
- B.7.1 RIF Register Mapping
- B.7.2 Transmit Data Register
- B.7.3 Receive Data Register – FFFF:7002 – XIO:0001
- B.7.4 Shift Data Registers (XSR and RSR)
- B.7.5 Control Register
- B.7.6 Control Register (SPCR) – XIO:0003

B.8 MPU Memory Interface—FFFF:FB00
- B.8.1 nCS0 Memory Range Register
- B.8.2 nCS1 Memory Range Register
- B.8.3 nCS2 Memory Range Register
- B.8.4 nCS3 Memory Range Register
- B.8.5 CS4 Memory Range Register
- B.8.6 nCS6 Internal Memory Range Register
- B.8.7 nCS7 Internal Memory Range Register
- B.8.8 API-TIPB Control Register
- B.8.9 Extracontrol Register

B.9 Memory Protection Unit (MPU)—FFFF:FF00
- B.9.1 Configuration Register Mapping

B.9.2 Protection Mode Definition
B.9.3 MPU Control Register Frame
B.9.4 Status Register
B.9.5 Control Register
B.9.6 Protection Mode Register
B.9.7 Base and Start Address Region n
B.9.8 End Address Definition, Region n
B.10 GEA Registers
B.10.1 GEA Register Mapping
B.10.2 Control Register (CNTL_REG) (Read/Write)—FFFF:C000
B.10.3 Status Register (STATUS_REG) (Read)—FFFF:C002
B.10.4 Interrupt Status Register (STATUS_IRQ_REG) (Read)—FFFF:C004
B.10.5 Uplink Configuration Registers (CONF_UL_REG (1:5))
B.10.6 Downlink Configuration Registers: CONF_DL_REG(1:5)
B.10.7 Ciphering Key Registers: KC_REG(1:4) Read/Write)—FFFF:C01A
B.10.8 FCS Uplink Registers: FCS_UL_REG(1:2)—FFFF:C022/C024
B.10.9 FCS Downlink Registers: FCS_DL_REG(1:2)—FFFF:C026/C028
B.10.10 Data Register
B.11 Configuration Registers—FFFE:F000
B.11.1 Configuration Register Mapping
B.11.2 Device ID Code Register
B.11.3 Device Version Code Register
B.11.4 cDSP ID Code Register
B.11.5 MPU ID Code Register
B.11.6 Die ID Code Register—FFFFE:F010 .. F016
B.11.7 DSP Configuration Register
B.11.8 Extended MPU Configuration Register
B.11.9 ASIC Configuration Register
B.11.10 IO Selection Register
B.11.11 MPU Software Trace Register
B.12 MCSI Registers—XIO:0800
B.12.1 MCSI Register Mapping
B.12.2 Control Registers—XIO:0803 .. 0805
B.12.3 Data Registers—XIO:0820..083F
B.13 MPU Interrupt Handler Registers—FFFF:FA00
B.13.1 IT Registers
B.13.2 Mask Interrupt Register
B.13.3 Source IRQ Binary-Coded Register
B.13.4 Source FIQ Binary Coded Register
B.13.5 Control Register
B.13.6 Interrupt Level Registers
B.13.7 Predefined Order in Case of Identical Priority Level
B.14 MPU to TIPB Registers—FFFF – F900
B.14.1 Interrupt Level Register
B.14.2 API-Write Status Register
B.14.3 MPU TIPB Control Register
B.14.4 Enhanced TIPB Control Register
B.15 DPLL Control Register—FFFF – 9800
B.15.1 DPLL Control Register (Read/Write)—FFFF:9800

B.16 MPUIO Registers—FFFE – 4800
B.16.1 Input Register
B.16.2 Output Register
B.16.3 Input/Output Control Register
B.16.4 MPUIO Control Register
B.16.5 Load Timer Register
B.16.6 Keyboard Row Input Register
B.16.7 Keyboard Column Output Register
B.16.8 Buzzer and Light Control Register
B.16.9 Light Power-Level Register
B.16.10 Buzzer Power-Level Register
B.16.11 GPIO Mode (GPIO_EVENT_MODE_REG) (Read/Write)—FFFE:4814
B.16.12 Keyboard/GPIO IRQ Register
B.16.13 Keyboard/GPIO Mask IRQ Register
B.16.14 GPIO Latch Register
B.16.15 Keyboard Interface
B.16.16 Pulse-Width Modulation (PWM)
B.17 SIM Registers—FFFE – 0000
B.17.1 SIM Control Register
B.17.2 SIM Status Register
B.17.3 SIM Configuration Register 1
B.17.4 SIM Configuration Register 2
B.17.5 SIM Interrupt Status Register
B.17.6 SIM Receive Byte Register
B.17.7 SIM Transmit Byte Register
B.17.8 SIM Interrupt Mask Register
B.17.9 SIM Card-Detect Interrupt Status Register
B.18 RTC Registers—FFFE – 1800
B.18.1 Time and Calendar Registers (TC)—FFFE:1800 .. 1806
B.18.2 TC Alarm Registers—FFFE:1808 .. 180D
B.18.3 General Registers—FFFE:1810 .. 1812
B.18.4 Compensation Registers—FFFE:1813 .. 1814
B.19 LPG Registers—FFFE – 7800
B.19.1 LPG Control Register
B.19.2 LPG Power Management Register
B.20 I2C Registers—FFFE – 2800
B.20.1 Device Register
B.20.2 Address Register
B.20.3 Date Write Register
B.20.4 Data Read Register
B.20.5 Command Register
B.20.6 FIFO Configuration Register
B.20.7 Clock Configuration Register
B.20.8 Clock-Configuration Functional Reference Register
B.20.9 Status FIFO Register
B.20.10 Activity Status Register
B.21 Timer Registers—FFFE – 3800/FFFE – 6800
B.21.1 TIMER1 Control Register—FFFE – 3800
B.21.2 Load TIMER1—FFFE – 3802
B.21.3 Read TIMER1—FFFE:3804

B.21.4 Control TIMER2—FFFE:6800
B.21.5 Load TIMER2—FFFE:6802
B.21.6 Read TIMER2—FFFE:6804
B.22 Watchdog Timer Registers—FFFF – F800
B.22.1 Watchdog Timer Control Register
B.22.2 Load Timer Register
B.22.3 Read Timer Register
B.22.4 Timer Mode Register
B.23 SPI Registers—FFFE – 3000
B.23.1 Setup Serial Port 1 Register
B.23.2 Setup Serial Port 2 Register
B.23.3 Serial Port Interface Control Register
B.23.4 Status Register
B.23.5 Transmit Registers
B.23.6 Receive Registers
B.24 µWire Registers—FFFE – 4000
B.24.1 Transmit Data Register
B.24.2 Receive Data Register
B.24.3 Control and Status Register
B.24.4 Setup Register 1
B.24.5 Setup Register 2
B.24.6 Setup Register 3
B.25 TSP Registers—FFFE – 0800
B.25.1 Parallel Bit Interface—0x06/0x07
B.25.2 TPU-Sequencer Internal Address Mapping—0x00/0x11
B.25.3 TSP Register Mapping
B.25.4 Transmit Registers (REG_TX_1/2/3/4)—0x02..0x05
B.25.5 Receive Registers—FFFE:0800 .. 0802
B.25.6 TSP Setup Registers—0x09 – 0x0B
B.25.7 Control Registers (REG_TSP_CTRL1/2)—0x00 – 0x01
B.25.8 TSP Gauging Enable Register—0x11
B.26 TPU Registers—FFFF – 1000
B.26.1 TPU RAM Memory Mapping—FFFF:9000
B.26.2 Control and Status Register
B.26.3 Interrupt Status Register
B.26.4 Interrupt Control Register
B.26.5 DSP Interrupt Occurrence Register
B.26.6 Offset Register
B.26.7 Synchro Register
B.27 ULPD Registers—FFFE – 2000
B.27.1 GSM Timer Registers—FFFE:2014 .. 201A
B.27.2 Setup Registers—FFFE:201C .. 2024
B.27.3 Gauging Registers—FFFE:2004 .. 2012
B.27.4 Counter Registers—FFFE:2008 .. 200E
B.27.5 GSM Timebase Registers: FFFE:2000 .. 2002
B.28 UART 16C750 Registers—FFFF – 5000/6000
B.28.1 UART/Modem Registers Mapping
B.28.2 UIR Mapping (UIR??)
B.28.3 Receiver Holding Register (RHR)
B.28.4 Transmit Holding Register (THR)

B.28.5 FIFO Control Register (FCR)
B.28.6 Supplementary Control Register (SCR)
B.28.7 Line Control Register (LCR)
B.28.8 Line Status Register (LSR)
B.28.9 Supplementary Status Register (SSR)
B.28.10 Modem Control Register (MCR)
B.28.11 Modem Status Register (MSR)
B.28.12 Interrupt Enable Register (IER)
B.28.13 Interrupt Identification Register (IIR)
B.28.14 Enhanced Feature Register (EFR)
B.28.15 Xon1/Addr1 Register
B.28.16 Xon2/Addr2 Register
B.28.17 Xoff1 Register
B.28.18 Xoff2 Register
B.28.19 Scratchpad Register (SPR)
B.28.20 Divisor Latches (DLL, DLH)
B.28.21 Transmission Control Register (TCR)
B.28.22 Trigger Level Register (TLR)
B.28.23 Mode Definition Register 1 (MDR1)
B.28.24 UART Autobauding Status Register (UASR) (UART/Modem Only)
B.28.25 UART Interface Register (UIR) (UART/Modem Only)
B.28.26 Mode Definition Register 2 (MDR2) (ART/IrDA Only)
B.28.27 Transmit Frame Length Registers (TXFLL, TXFLH) (UART/IrDA Only)
B.28.28 Received Frame Length Register (RXFLL, RXFLH) (ART/IrDA Only)
B.28.29 Status FIFO Line Status Register (SFLSR) (ART/IrDA Only)
B.28.30 RESUME Register (UART/IrDA Only)
B.28.31 Status FIFO Register (SFREGL, SFREGH) (ART/IrDA Only)
B.28.32 BOF Length Register (BLR) (UART/IrDA Only)
B.28.33 DIV1.6 Register (UART/IrDA Only)
B.28.34 Auxiliary Control Register (ACREG) (UART/IrDA Only)

C Pin Descriptions
C.1 OMAP730 Platform Pins

D Packaging
D.1 Package Pin Location
D.2 Mechanical Data

E Peripherals Revision Number
E.1 Peripherals Revision Number

Appendix D

OMAP730–OMAP850 Differences

This appendix presents the differences between the OMAP730 and the OMAP850 devices by providing an overview of OMAP850 features for OMAP730 knowledgeable users.

Topic **Page**

D.1 OMAP850/OMAP730 Comparison Overview D-2

D.1 OMAP850/OMAP730 Comparison Overview

The Texas Instruments OMAP850 multimedia processor is a close derivative of OMAP730 processor. OMAP850 is an improved platform for key performance aspects. OMAP850 also receives some changes in MPU-S peripheral features list and pin multiplex. High compatibility level is preserved in both hardware and software, as follows:

- Package
  - Identical package: 289 ball grid array (BGA)
  - Identical package pins location: pins name and pins assignment to balls
  - Compatible pin multiplex (with few exceptions, see removed features below)
  - Pin multiplex additions for new features and enhanced support
- GSM-S features
  - GPRS-EDGE feature
  - Binary software compatibility
- MPU-S enhanced features
  - Optimized internal SRAM interface for higher frame-buffer performances
  - Optimized OMAP3.2 core for DDR and DMA performances
  - 8-bit parallel camera interface data throughput improvement (same camera features; camera support moved from MPU-S peripheral bus to MPU-S TC OCP-T interface) for bigger sensor/higher frame-rate
- MPU-S removed features
  - CompactFlash interface not supported anymore
  - GPIO_72/73/74 removed from pin multiplex on DQSH/DQSL/SDCLKX DDR pins (DDR timings constraints)
- MPU-S new features
  - Addition of DDR memory support to SDRAM
  - Addition of external DMA request line support for EMIFS components
  - Addition of SHPLCD for glueless interfacing of LCD controller to Sharp TFT-LCD panels
  - Addition of traffic controller burst doubler feature in OMAP3.2 core for ARM926EJS cache-fill performance improvement