AM5K2E04/02

Multicore ARM KeyStone II System-on-Chip (SoC)

Data Manual

PRODUCT PREVIEW information applies to products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

Literature Number: SPRS864
November 2012

Release History

Release	Date	Description/Comments
SPRS864	November 2012	Initial Release

Contents

List of Figures

List of Tables

1 AM5K2E04/02 Features and Description

1.1 Features

- ARM® Cortex™-A15 MPCore™ CorePac
  - Up to Four 1.4-GHz ARM Cortex-A15 Processor Cores
  - 4MB L2 Cache Memory Shared by all Cortex-A15 Processor Cores
  - Full Implementation of ARMv7-A Architecture Instruction Set
  - 32KB L1 Instruction and Data Caches per Core
  - AMBA 4.0 AXI Coherency Extension (ACE) Master Port, Connected to MSMC (Multicore Shared Memory Controller) for Low Latency Access to SRAM and DDR3
- Multicore Shared Memory Controller (MSMC)
  - 2 MB SRAM Memory Shared by ARM CorePac
  - Memory Protection Unit for Both SRAM and DDR3_EMIF
- Multicore Navigator
  - 8k Multi-Purpose Hardware Queues with Queue Manager
  - One Packet-Based DMA Engine for Zero-Overhead Transfers
- Two Network Coprocessors Each with
  - Packet Accelerator Enables Support for
    - Transport Plane IPsec, GTP-U, SCTP, PDCP
    - L2 User Plane PDCP (RoHC, Air Ciphering)
    - 1 Gbps Wire Speed Throughput at 1.5 MPackets Per Second
  - Security Accelerator Engine Enables Support for
    - IPSec, SRTP, 3GPP and WiMAX Air Interface, and SSL/TLS Security
    - ECB, CBC, CTR, F8, A5/3, CCM, GCM, HMAC, CMAC, GMAC, AES, DES, 3DES, Kasumi, SNOW 3G, SHA-1, SHA-2 (256-bit Hash), MD5
    - Up To 6.4 Gbps IPSec and 3 Gbps Air Ciphering
  - Ethernet Subsystem
    - Four SGMII Ports with Wire Rate Switching
    - IEEE1588 v2 (with Annex D/E/F) Support
- Peripherals
  - Two PCIe Gen2 Controllers with Support for
    - Two Lanes per Controller
    - Supports Up To 5 GBaud
  - One Hyperlink
    - Supports Connections to Other KeyStone II Architecture Devices Providing Resource Scalability
    - Supports Up To 50 GBaud
  - 10-Gigabit Ethernet (10-GbE) Switch Subsystem
    - Two SGMII/XFI Ports with Wire Rate Switching and MACSEC Support
    - IEEE1588 v2 (with Annex D/E/F) Support
  - One 72-Bit DDR3/DDR3L Interface with Speeds Up To 1600 MTPS in DDR3 Mode
  - EMIF16 Interface
  - Two USB 2.0/3.0 Controllers
  - Two UART Interfaces
  - Three I^2C Interfaces
  - 32 GPIO Pins
  - Three SPI Interfaces
  - One TSIP
- System Resources
  - Three On-Chip PLLs
  - SmartReflex Automatic Voltage Scaling
  - Semaphore Module
  - Twelve 64-Bit Timers
  - Five Enhanced Direct Memory Access (EDMA) Modules
- Commercial Case Temperature:
  - 0°C to 85°C
- Extended Case Temperature:
  - - 40°C to 100°C

PRODUCT PREVIEW

1.2 KeyStone II Architecture

TI's KeyStone II Multicore Architecture provides a unified platform for integrating RISC and DSP processing cores along with both hardware/firmware based application-specific acceleration and high performance I/Os. The KeyStone II Multicore Architecture is a proven device architecture to achieve the full performance entitlement through the following major components: TeraNet, Multicore Shared Memory Controller, Multicore Navigator, and HyperLink.

TeraNet is a multipoint to multipoint non-blocking switch fabric. Its distributed arbiter provides multiple duplex communication channels in parallel between the master and slave ports without interference. The priority based arbitration mechanism ensures the delivery of the critical traffic delivery in the system.

The Multicore Shared Memory Controller (MSMC) is the center of the KeyStone II memory architecture. It provides multiple fast and high-bandwidth channels for processor cores to access DDR and minimizes the access latency by directly connecting to the DDR. The MSMC also provides the flexibility to expand processor cores with little impact at the device level. In addition, it provides multi-bank based fast on-chip SRAM shared among processor cores and IOs. It also provides the I/O cache coherency for the device when the Cortex-A15 processor core is integrated.

The Multicore Navigator provides a packet-based IPC mechanism among processing cores and packet based peripherals. The hardware-managed queues supports multiple-in-multiple-out mode without using mutex. Coupled with the packet-based DMA, the Multicore Navigator provides a highly efficient and software-friendly tool to offload the processing core to achieve other critical tasks.

HyperLink provides a 50-GBaud chip-level interconnect that allows devices to work in tandem. Its low latency, low overhead and high throughput makes it an ideal interface for chip-to-chip interconnections.

There are two generations of KeyStone architecture. The AM5K2E04/02 is based on KeyStone II, which integrates a Cortex-A15 processor CorePac.

1.3 Device Description

The AM5K2E04/02 is a high performance device based on TI's KeyStone II Multicore SoC Architecture, incorporating the most performance-optimized Cortex-A15 processor dual-core or quad-core CorePac that can run at a core speed of up to 1.4 GHz. TI's AM5K2E04/02 device enables a high performance, power-efficient and easy to use platform for developers of a broad range of applications such as enterprise grade networking end equipment, data center networking, mission critical, medical imaging, test and automation.

TI's KeyStone II Architecture provides a programmable platform integrating various subsystems (e.g., ARM CorePac (Cortex-A15 Processor Quad Core CorePac), network processing, and uses a queue-based communication system that allows the device resources to operate efficiently and seamlessly. This unique device architecture also includes a TeraNet switch that enables the wide mix of system elements, from programmable cores to high-speed IO, to each operate at maximum efficiency with no blocking or stalling.

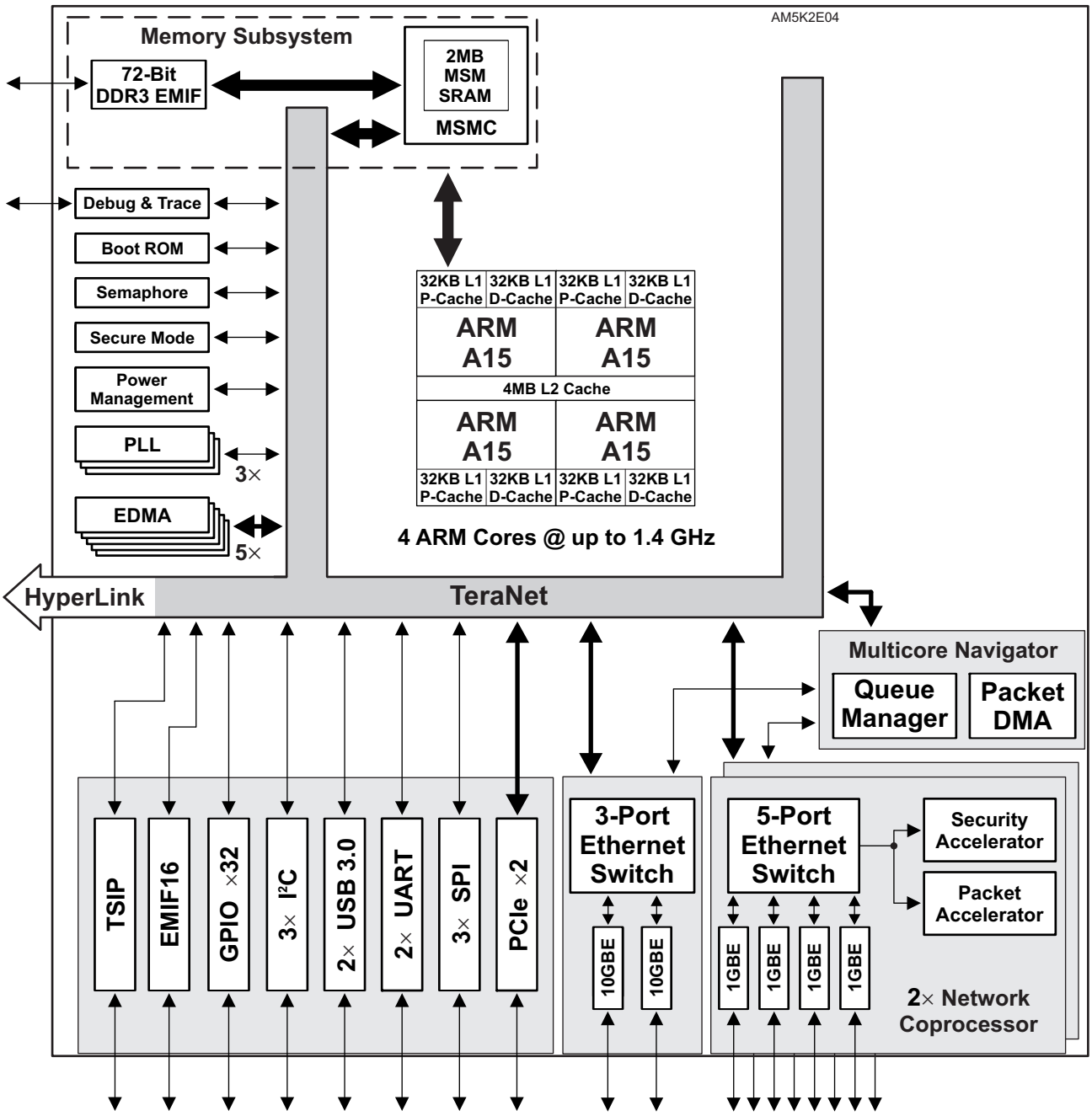
The AM5K2E04/02 KeyStone II device integrates a large amount of on-chip memory. The Cortex-A15 processor cores each have 32KB of L1Data and 32KB of L1 Instruction cache. The up to four Cortex A15 cores in the ARM CorePac share a 4MB L2 Cache. The device also integrates 2MB of Multicore Shared Memory (MSMC) that can be used as a shared L3 SRAM. All L2 and MSMC memories incorporate error detection and error correction. For fast access to external memory, this device includes a 64-bit DDR-3 (72-bit with ECC support) external memory interface (EMIF) running at 1600 MTPS.

The device enables developers to use a variety of development and debugging tools that include GNU GCC, GDB, Open source Linux, Eclipse based debugging environment enabling kernel and user space debugging using a variety of Eclipse plug-ins including TI's industry leading IDE Code Composer Studio.

1.4 Functional Block Diagram

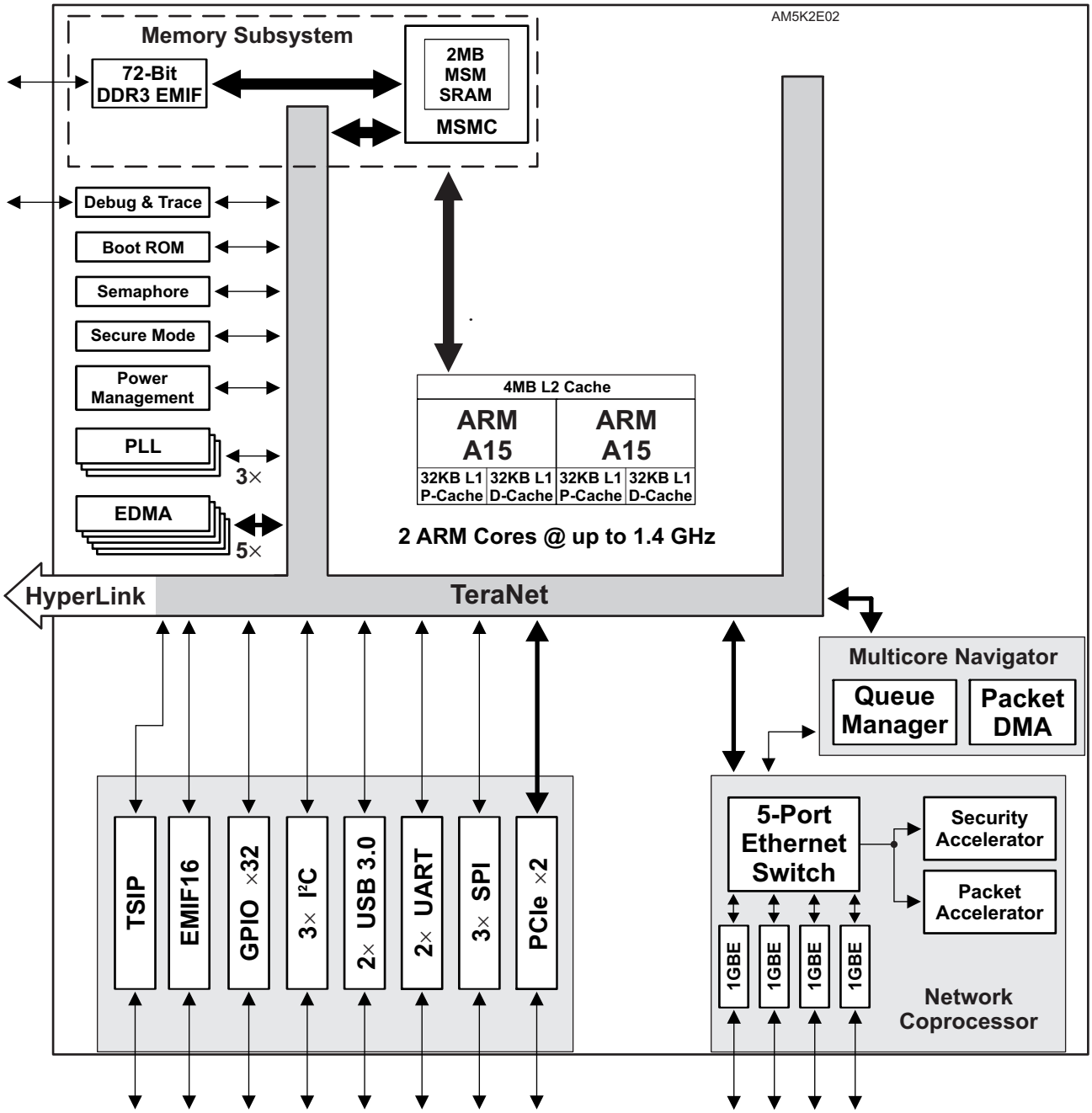
The figures below show the functional block diagrams of the AM5K2E04/02 devices.

Figure 1-1 AM5K2E04 Functional Block Diagram



PRODUCT PREVIEW

Figure 1-2 AM5K2E02 Functional Block Diagram



PRODUCT PREVIEW

1.5 Device Characteristics

Table 1-1 provides an overview of the AM5K2E04/02 device. The table shows the significant features of the device, including the capacity of on-chip RAM, the peripherals, the CPU frequency, and the package type with pin count.

Table 1-1 Characteristics of the AM5K2E04/02 Processor

HARDWARE FEATURES		AM5K2E02	AM5K2E04
ARM Cores	ARM Cortex A15 Cores	2	4
	ARM L1 instruction cache memory size (per core)	32KB	
	ARM L1 data cache memory size (per core)	32KB	
	ARM L2 unified cache memory size (shared by all cores)	4MB	
Peripherals	DDR3 memory controller (72-bit bus width) [1.5 V/1.35V] (clock source = DDRREFCLKN|P)	1	
	EDMA3 (64 independent channels) [CPU/3 clock rate]	5	
	Hyperlink	1	
	USB 3.0	2	
	I^2C	3	
	SPI	3	
	PCIe (2 lanes per instance)	2	
	UART	2	
	10/100/1000/10000 Ethernet ports	0	2
	10/100/1000 Ethernet ports	4	8
	Management Data Input/Output (MDIO)	3	
	64-bit timers (configurable)	Twelve 64-bit ***or*** Twenty four 32-bit	
	General-Purpose Input/Output port (GPIO)	32	
	TSIP	1	
Accelerators	Packet Accelerator	1	2
	Security Accelerator (1)	1	2
On-Chip L3 Memory	Organization	2MB MSM SRAM 256 KB L3 ROM	
JTAG BSDL_ID	JTAGID Register (address location: 0x02620018)	TBD	
Frequency	MHz	1400 (1.4 GHz) [-1400] 1200 (1.2 GHz) [-1200] 1000 (1.0 GHz) [-1000]	
Cycle Time	ns	0.71 ns [-1400] 0.83 ns [-1200] 1 ns [-1000]	
Voltage	Core (V)	SmartReflex variable supply	
	I/O (V)	1.35 V, 1.5 V, 1.8 V, and 3.3 V	
BGA Package	TBD	TBD	
Process Technology	nm	28 nm	
Product Status (2)	Product Preview (PP), Advance Information (AI), or Production Data (PD)	PP	
End of Table 1-1			

1 The Security Accelerator function is subject to export control and will be enabled *only* for approved device shipments.
2 .PRODUCT PREVIEW information applies to products in the formative or design phase of development. Characteristic data and other specifications are design goals. Texas Instruments reserves the right to change or discontinue these products without notice.

PRODUCT PREVIEW

1.6 ARM CorePac

The ARM CorePac of the AM5K2E04/02 integrates a Cortex-A15 Cluster (4 Cortex™-A15 processors) with additional logic for bus protocol conversion, emulation, interrupt handling, and debug related enhancements. The Cortex™-A15 processor is an ARMv7A-compatible, multi-issue out-of-order, superscalar pipeline with integrated L1 caches. The implementation also supports advanced SIMDV2 (Neon technology) and VFPv4 (Vector Floating Point) architecture extensions, security, virtualization, LPAE (Large Physical Address Extension), and multiprocessing extensions. The quad core cluster includes a 4MB L2 cache and support for AMBA4 AXI and AXI Coherence Extension (ACE) protocols. For more information see the ARM CorePac User Guide for KeyStone II Devices User Guide listed in 1.8 "Related Documentation from Texas Instruments" on page 13.

1.7 Development Tools

1.7.1 Development Support

In case the customer would like to develop their own features and software on the AM5K2E04/02 device, TI offers an extensive line of development tools for the KeyStone II platform, including tools to evaluate the performance of the processors, generate code, develop algorithm implementations, and fully integrate and debug software and hardware modules. The tool's support documentation is electronically available within the Code Composer Studio™ Integrated Development Environment (IDE).

The following products support development of Cortex-A15 processor-based applications:

- **Software Development Tools:**
  - TBD
- **Hardware Development Tools:**
  - TBD

1.7.2 Device and Development-Support Tool Nomenclature

To designate the stages in the product development cycle, TI assigns prefixes to the part numbers of all devices and support tools. Each family member has one of two prefixes: X or [blank]. These prefixes represent evolutionary stages of product development from engineering prototypes through fully qualified production devices/tools.

Device development evolutionary flow:

- **X:** Experimental device that is not necessarily representative of the final device's electrical specifications
- **[Blank]:** Fully qualified production device

Support tool development evolutionary flow:

- **X:** Development-support product that has not yet completed Texas Instruments internal qualification testing.
- **[Blank]:** Fully qualified development-support product

Experimental (X) and fully qualified [Blank] devices and development-support tools are shipped with the following disclaimer:

Developmental product is intended for internal evaluation purposes.

Fully qualified and production devices and development-support tools have been characterized fully, and the quality and reliability of the device have been demonstrated fully. TI's standard warranty applies.

Predictions show that experimental devices (X) have a greater failure rate than the standard production devices. Texas Instruments recommends that these devices not be used in any production system because their expected end-use failure rate still is undefined. Only qualified production devices are to be used.

PRODUCT PREVIEW

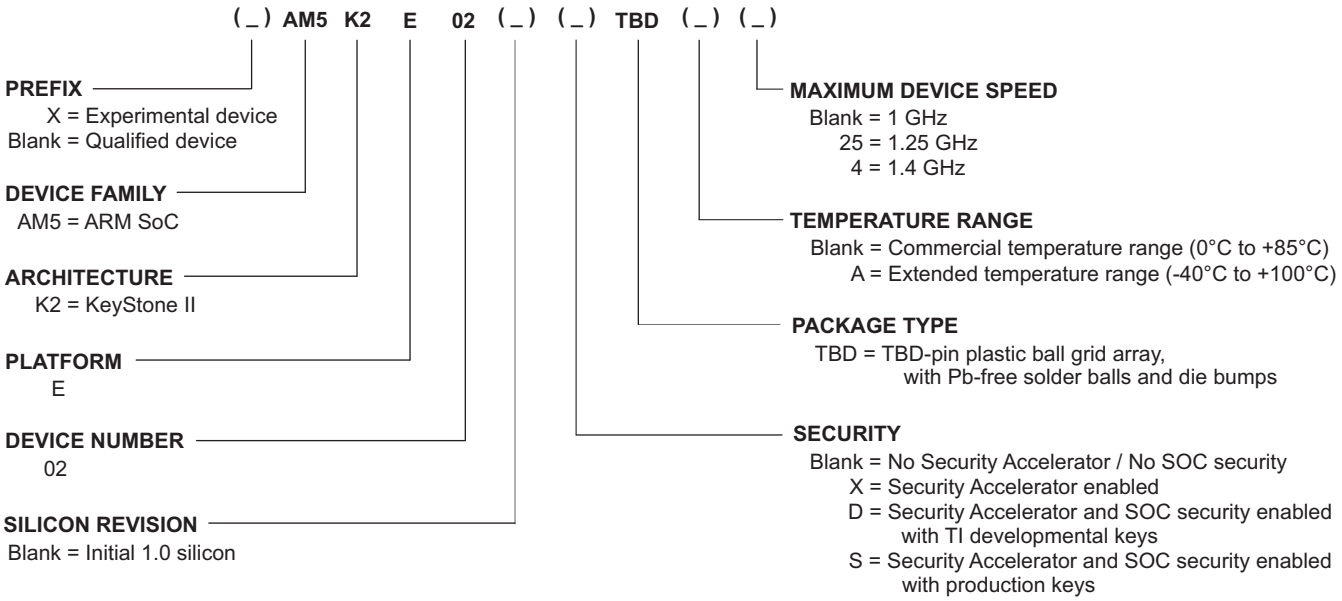
TI device nomenclature also includes a suffix with the device family name. This suffix indicates the package type (for example, AAW), the temperature range (for example, blank is the default case temperature range), and the device speed range, in Megahertz (for example, blank is 1000 MHz [1 GHz]).

For device part numbers and further ordering information for AM5K2E04/02 in the AAW package type, see the TI website www.ti.com or contact your TI sales representative.

1.7.3 Device Nomenclature

The following figures provide a legend for reading the complete device name for a KeyStone II device.

Figure 1-3 Device Nomenclature for AM5K2E02



PRODUCT PREVIEW

1.8 Related Documentation from Texas Instruments

These documents describe the AM5K2E04/02 Multicore ARM KeyStone II System-on-Chip (SoC). Copies of these documents are available on the Internet at www.ti.com.

64-bit Timer (Timer 64) for KeyStone Devices User Guide	SPRUGV5
ARM CorePac User Guide for KeyStone II Devices User Guide	SPRUHJ4
Chip Interrupt Controller (CIC) for KeyStone Devices User Guide	SPRUGW4
Debug and Trace for KeyStone Devices User Guide	SPRUGZ2
DDR3 Memory Controller for KeyStone Devices User Guide	SPRUGV8
External Memory Interface (EMIF16) for KeyStone Devices User Guide	SPRUGZ3
Emulation and Trace Headers Technical Reference	SPRU655
Enhanced Direct Memory Access 3 (EDMA3) for KeyStone Devices User Guide	SPRUGS5
General Purpose Input/Output (GPIO) for KeyStone Devices User Guide	SPRUGV1
Gigabit Ethernet (GbE) Switch Subsystem (1 GB) for KeyStone Devices User Guide	SPRUGV9
Gigabit Ethernet (GbE) Switch Subsystem (10 GB) for KeyStone II Devices User Guide	SPRUHJ5
Hyperlink for KeyStone Devices User Guide	SPRUGW8
Inter Integrated Circuit (I^2C) for KeyStone Devices User Guide	SPRUGV3
Interrupt Controller (INTC) for KeyStone Devices User Guide	SPRUGW4
Memory Protection Unit (MPU) for KeyStone Devices User Guide	SPRUGW5
Multicore Navigator for KeyStone Devices User Guide	SPRUGR9
Multicore Shared Memory Controller (MSMC) for KeyStone II Devices User Guide	SPRUHJ6
Network Coprocessor (NETCP) for KeyStone Devices User Guide	SPRUGZ6
Optimizing Application Software on KeyStone Devices	SPRABG8
Packet Accelerator (PA) for KeyStone Devices User Guide	SPRUGS4
Peripheral Component Interconnect Express (PCIe) for KeyStone Devices User Guide	SPRUGS6
Phase Locked Loop (PLL) Controller for KeyStone Devices User Guide	SPRUGV2
Power Sleep Controller (PSC) for KeyStone Devices User Guide	SPRUGV4
Security Accelerator (SA) for KeyStone Devices User Guide	SPRUGY6
Semaphore2 Hardware Module for KeyStone Devices User Guide	SPRUGS3
Serial Peripheral Interface (SPI) for KeyStone Devices User Guide	SPRUGP2
Telecom Serial Interface Port (TSIP) for the C66x DSP User Guide	SPRUGY4
Universal Serial Bus 3 (USB3) for KeyStone II Devices User Guide	SPRUHJ7
Universal Asynchronous Receiver/Transmitter (UART) for KeyStone Devices User Guide	SPRUGP1

PRODUCT PREVIEW

2 ARM CorePac

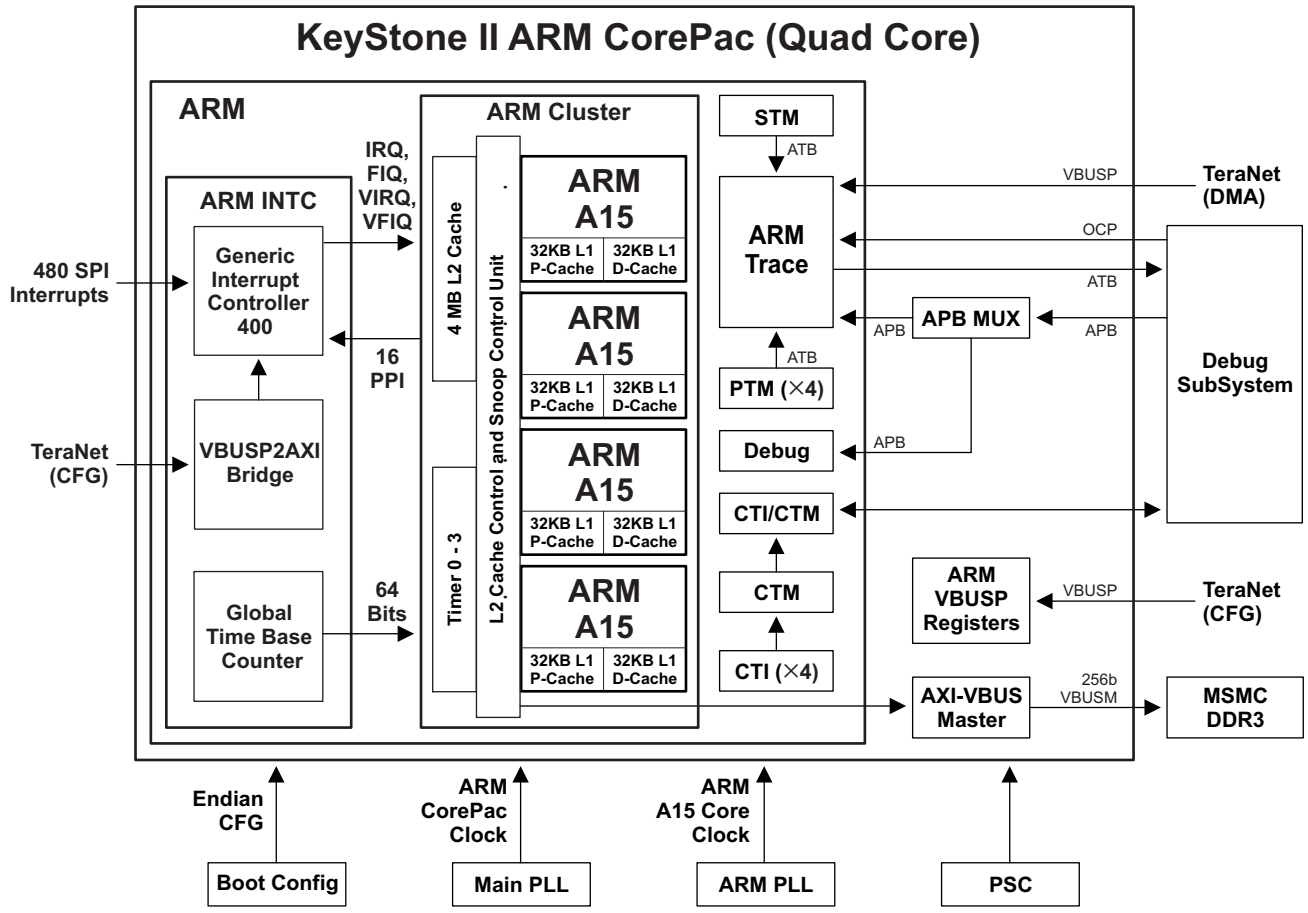
The ARM CorePac is added in the AM5K2E04/02 to enable the ability for layer 2 and layer 3 processing on-chip. Operations such as traffic control, local O&M, NBAP/FP termination, and SCTP processing can all be performed with the Cortex-A15 processor core.

The ARM CorePac of the AM5K2E04/02 integrates a Cortex-A15 processor cluster (four Cortex™-A15 processors) with additional logic for bus protocol conversion, emulation, interrupt handling, and debug related enhancements. The Cortex™-A15 processor is an ARMv7A-compatible, multi-issue out-of-order superscalar execution engine with integrated L1 caches. The implementation also supports advanced SIMDv2 (NEON™ technology) and VFPv4 (vector floating point) architecture extensions, security, virtualization, LPAE (large physical address extension), and multiprocessing extensions. The ARM CorePac includes a 4MB L2 cache and support for AMBA4 AXI and AXI coherence extension (ACE) protocols. An interrupt controller is included in the ARM CorePac to handle host interrupt requests in the system.

The ARM CorePac has three functional clock domains, including a high-frequency clock domain used by the Cortex™-A15. The high-frequency domain is isolated from the rest of the device by asynchronous bridges.

Figure 2-1 shows an overall view of the Quad ARM CorePac.

Figure 2-1 KeyStone II ARM CorePac Block Diagram



PRODUCT PREVIEW

2.1 Features

The key features of the Quad Core ARM CorePac are as follows:

- Four Cortex-A15 processors
  - Cortex-A15 processor revision R2P4.
  - ARM architecture version 7 ISA.
  - Multi-issue, out-of-order, superscalar pipeline.
  - L1 and L2 instruction and data cache of 32 KB, 4-way, 16 word line with 128 bit interface.
  - Integrated L2 cache of 4MB, 8-way, 16 word line, 128-bit interface to L1 along with ECC/parity.
  - Includes the NEON media coprocessor (NEON™), which implements the advanced SIMDv2 media processing architecture and the VFPv4 Vector Floating Point architecture.
  - The external interface uses the AXI protocol configured to 128-bit data width.
  - Includes the System Trace Macrocell (STM) support for non-invasive debugging.
  - Implements the ARMv7 debug with watchpoint and breakpoint registers and 32-bit advanced peripheral bus (APB) slave interface to CoreSight™ debug systems.
- Interrupt controller
  - Supports up to 480 interrupt requests
- Emulation/debug
  - Compatible with CoreSight™ architecture
- Clock generation
  - Through the dedicated ARM PLL

2.2 System Integration

The ARM CorePac integrates the following group of submodules.

- **Cortex™-A15 Processors:** Provides a high processing capability, including the NEON™ technology for mobile multimedia acceleration. The Cortex™-A15 communicates with the rest of the ARM CorePac through an AXI bus with an AXI2VBUSM bridge and receives interrupts from the ARM CorePac interrupt controller (ARM INTC).
- **Interrupt Controller:** Handles interrupts from modules outside of the ARM CorePac (for details, see "ARM Interrupt Controller").
- **Clock Divider:** Provides the required divided clocks to the internal modules of the ARM CorePac and has a clock input from the ARM PLL and the Main PLL
- **In-Circuit Emulator:** Fully compatible with CoreSight™ architecture and enables debugging capabilities.

2.3 ARM Cortex-A15 Processor

2.3.1 Overview

The ARM Cortex™-A15 processor incorporates the technologies available in the ARM7™ architecture. These technologies include NEON™ for media and signal processing and Jazelle™ RCT for acceleration of real-time compilers, Thumb®-2 technology for code density, and the VFPv4 floating point architecture. For details, see the ARM Cortex™-A15 Processor Technical Reference Manual.

2.3.2 Features

Table 2-1 shows the features supported by the Cortex-A15 processor core.

Table 2-1 Cortex-A15 Processor Core Supported Features

Features	Description
ARM version 7-A ISA	Standard Cortex-A15 processor instruction set + Thumb2, ThumbEE, JazelleX Java accelerator, and media extensions
	Backward compatible with previous ARM ISA versions
Cortex-A15 processor version	R2P4
Integer core	Main core for processing integer instructions
NEON core	Gives greatly enhanced throughput for media workloads and VFP-Lite support
Architecture Extensions	Security, virtualization and LPAE (40bit virtual address) extensions
L1 Lcache and Dcache	32KB, 4-way, 16 word line, 128 bit interface
L2 cache	4096KB, 8-way, 16 word line, 128 bit interface to L1, ECC/Parity is supported shared between cores
	L2 valid bits cleared by software loop or by hardware
Cache Coherency	Support for coherent memory accesses between A15 cores and other non-core master peripherals (Ex: EDMA) in the DDR3A and MSMC SRAM space. (Cache coherency is not supported between Cortex-A15 and DSP cores or between DSP cores.)
Branch target address cache	Dynamic branch prediction with Branch Target Buffer (BTB) and Global History Buffer (GHB), a return stack, and an indirect predictor
Enhanced memory management unit	Mapping sizes are 4KB, 64KB, 1MB, and 16MB
Buses	128b AXI4 internal bus from Cortex-A15 converted to a 256b VBUSM to interface (through the MSMC) with MSMC SRAM, DDR EMIF, ROM, Interrupt controller and other system peripherals
Non-invasive Debug Support	Processor instruction trace using 4x Program Trace Macrocell (Coresight™ PTM), Data trace (print-f style debug) using System Trace Macrocell (Coresight™ STM) and Performance Monitoring Units (PMU)
Misc Debug Support	JTAG based debug and Cross triggering
Clocking	Dedicated ARM PLL for flexible clocking scenarios
Voltage	Dedicated SmartReflex voltage domain for automatic voltage scaling
Power	Support for standby modes and separate core power domains for additional leakage power reduction
End of Table 2-1	

PRODUCT PREVIEW

2.3.3 ARM Interrupt Controller

The ARM CorePac interrupt controller (AINTC) is responsible for prioritizing all service requests from the system peripherals and the Secondary interrupt controller INCT2 and then generating either nIRQ or nFIQ to the Cortex-A15 processor. The type of the interrupt (nIRQ or nFIQ) and the priority of the interrupt inputs are programmable. The AINTC interfaces to the Cortex-A15 processor via the AXI port through an VBUS2AXI bridge and runs at half the processor speed. It has the capability to handle up to 480 requests, which can be steered/prioritized as A15 nFIQ or nIRQ interrupt requests.

The general features of the AINTC are:

- Up to 480 level sensitive shared peripheral interrupts (SPI) inputs
- Individual priority for each interrupt input
- Each interrupt can be steered to nFIQ or nIRQ
- Independent priority sorting for nFIQ and nIRQ
- Secure mask flag

On the chip level, there is a dedicated chip level interrupt controller to serve the ARM interrupt controller. See the Interrupt section for more details.

The figures below show an overall view of the ARM CorePac interrupt controller.

Figure 2-2 ARM Interrupt Controller for 2 Cortex-A15 Processor Cores

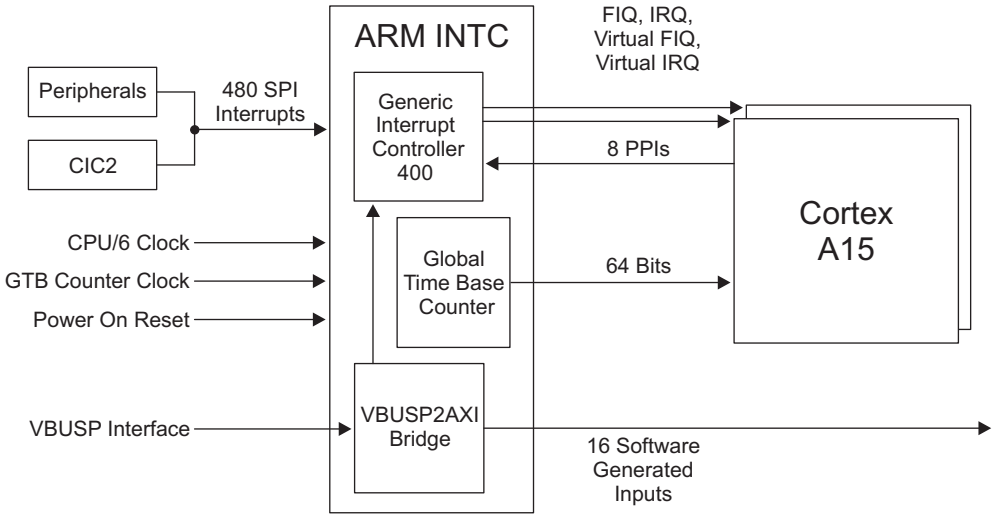
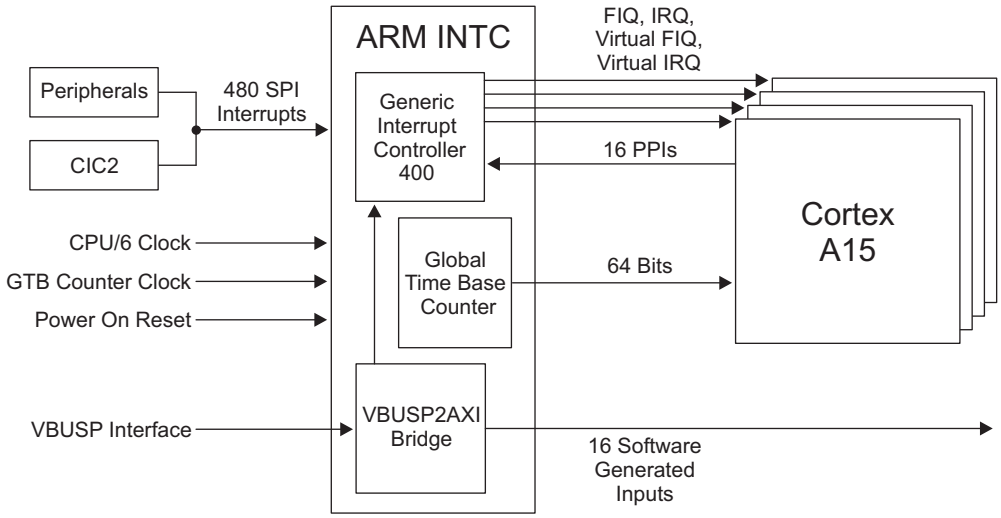


Figure 2-3 ARM Interrupt Controller for 4 Cortex-A15 Processor Cores



2.3.4 Endianess

The ARM CorePac can operate in either little endian or big endian mode. When the ARM CorePac is in little endian mode and the rest of the system is in big endian mode, the bridges in the ARM CorePac are responsible for performing the endian conversion.

2.4 CFG Connection

The ARM CorePac has two slave ports. The AM5K2E04/02 masters cannot access the ARM CorePac internal memory space.

1. Slave port 0 (TeraNet 3P_A) is a 32 bit wide port used for the ARM Trace module.
2. Slave port 1 (TeraNet 3P_B) is a 32 bit wide port used to access the rest of the system configuration.

2.5 Main TeraNet Connection

There is one master port coming out of the ARM CorePac. The master port is a 256 bit wide port for the transactions going to the MSMC and DDR_EMIF data spaces.

2.6 Clocking and Reset

2.6.1 Clocking

The ARM CorePac includes a dedicated embedded DPLL (ARM PLL). The Cortex-A15 processor core clocks are sourced from this ARM PLL Controller. The main Cortex-A15 processor core clock has a maximum frequency of 1.4 Ghz. A clock divider from the main PLL clock source (/1, /3 and /6) is used for deriving the clocks for other Cortex-A15 processor internal modules. All major modules inside the ARM CorePac are clocked at half the frequency of the Cortex-A15 processor core, such as AINTC and CoreSight Debug System modules. The emulation clock within the Cortex-A15 processor core runs at one third the frequency of the Cortex-A15 processor core. The divider of the output clock is programmable, with the frequency relative to the Cortex-A15 processor core.

2.6.2 Reset

The ARM CorePac does not support local reset. It is reset whenever the device is under reset. In addition, the interrupt controller (AINTC) can only be reset during $\overline{\text{POR}}$ and $\overline{\text{RESETFULL}}$.

PRODUCT PREVIEW

3 Memory, Interrupts, and EDMA for AM5K2E04/02

3.1 Memory Map Summary for AM5K2E04/02

The following table shows the memory map address ranges of the device.

Table 3-1 Device Memory Map Summary for AM5K2E04/02 (Part 1 of 12)

Physical 40 bit Address		Bytes	ARM View	SOC View
Start	End			
00 0000 0000	00 0003 FFFF	256K	ARM ROM	ARM ROM
00 0004 0000	00 007F FFFF	8M-256K	Reserved	Reserved
00 0080 0000	00 008F FFFF	1M	Reserved	Reserved
00 0090 0000	00 00DF FFFF	5M	Reserved	Reserved
00 00E0 0000	00 00E0 7FFF	32K	Reserved	Reserved
00 00E0 8000	00 00EF FFFF	1M-32K	Reserved	Reserved
00 00F0 0000	00 00F0 7FFF	32K	Reserved	Reserved
00 00F0 8000	00 00FF FFFF	1M-32K	Reserved	Reserved
00 0100 0000	00 0100 FFFF	64K	ARM AXI2VBUSM registers	Reserved
00 0101 0000	00 010F FFFF	1M-64K	Reserved	Reserved
00 0110 0000	00 0110 FFFF	64K	ARM STM Stimulus Ports	Reserved
00 0101 0000	00 01BF FFFF	11M-64K	Reserved	Reserved
00 01C0 0000	00 01CF FFFF	1M	Reserved	Reserved
00 01D0 0000	00 01D0 007F	128	Tracer CFG0	Tracer CFG0
00 01D0 0080	00 01D0 7FFF	32K-128	Reserved	Reserved
00 01D0 8000	00 01D0 807F	128	Tracer CFG1	Tracer CFG1
00 01D0 8080	00 01D0 FFFF	32K-128	Reserved	Reserved
00 01D1 0000	00 01D1 007F	128	Tracer CFG2	Tracer CFG2
00 01D1 0080	00 01D1 7FFF	32K-128	Reserved	Reserved
00 01D1 8000	00 01D1 807F	128	Tracer CFG3	Tracer CFG3
00 01D1 8080	00 01D1 FFFF	32K-128	Reserved	Reserved
00 01D2 0000	00 01D2 007F	128	Tracer CFG4	Tracer CFG4
00 01D2 0080	00 01D2 7FFF	32K-128	Reserved	Reserved
00 01D2 8000	00 01D2 807F	128	Tracer CFG5	Tracer CFG5
00 01D2 8080	00 01D2 FFFF	32K-128	Reserved	Reserved
00 01D3 0000	00 01D3 007F	128	Tracer CFG6	Tracer CFG6
00 01D3 0080	00 01D3 7FFF	32K-128	Reserved	Reserved
00 01D3 8000	00 01D3 807F	128	Tracer CFG7	Tracer CFG7
00 01D3 8080	00 01D3 FFFF	32K-128	Reserved	Reserved
00 01D4 0000	00 01D4 007F	128	Tracer CFG8	Tracer CFG8
00 01D4 0080	00 01D4 7FFF	32K-128	Reserved	Reserved
00 01D4 8000	00 01D4 807F	128	Tracer CFG9	Tracer CFG9
00 01D4 8080	00 01D4 FFFF	32K-128	Reserved	Reserved
00 01D5 0000	00 01D5 007F	128	Reserved	Reserved
00 01D5 0080	00 01D5 7FFF	32K-128	Reserved	Reserved
00 01D5 8000	00 01D5 807F	128	Reserved	Reserved
00 01D5 8080	00 01D5 FFFF	32K-128	Reserved	Reserved
00 01D6 0000	00 01D6 007F	128	Reserved	Reserved
00 01D6 0080	00 01D6 7FFF	32K-128	Reserved	Reserved

Table 3-1 Device Memory Map Summary for AM5K2E04/02 (Part 2 of 12)

Physical 40 bit Address		Bytes	ARM View	SOC View
Start	End			
00 01D6 8000	00 01D6 807F	128	Reserved	Reserved
00 01D6 8080	00 01D6 FFFF	32K-128	Reserved	Reserved
00 01D7 0000	00 01D7 007F	128	Reserved	Reserved
00 01D7 0080	00 01D7 7FFF	32K-128	Reserved	Reserved
00 01D7 8000	00 01D7 807F	128	Reserved	Reserved
00 01D7 8080	00 01D7 FFFF	32K-128	Reserved	Reserved
00 01D8 0000	00 01D8 007F	128	Reserved	Reserved
00 01D8 0080	00 01D8 7FFF	32K-128	Reserved	Reserved
00 01D8 8000	00 01D8 807F	128	Reserved	Reserved
00 01D8 8080	00 01D8 8FFF	32K-128	Reserved	Reserved
00 01D9 0000	00 01D9 007F	128	Reserved	Reserved
00 01D9 0080	00 01D9 7FFF	32K-128	Reserved	Reserved
00 01D9 8000	00 01D9 807F	128	Reserved	Reserved
00 01D9 8080	00 01D9 FFFF	32K-128	Reserved	Reserved
00 01DA 0000	00 01DA 007F	128	Tracer CFG20	Tracer CFG20
00 01DA 0080	00 01DA 7FFF	32K-128	Reserved	Reserved
00 01DA 8000	00 01DA 807F	128	Reserved	Reserved
00 01DA 8080	00 01DA FFFF	32K-128	Reserved	Reserved
00 01DB 0000	00 01DB 007F	128	Tracer CFG22	Tracer CFG22
00 01DB 0080	00 01DB 7FFF	32K-128	Reserved	Reserved
00 01DB 8000	00 01DB 807F	128	Reserved	Reserved
00 01DB 8080	00 01DB 8FFF	32K-128	Reserved	Reserved
00 01DC 0000	00 01DC 007F	128	Tracer CFG24	Tracer CFG24
00 01DC 0080	00 01DC 7FFF	32K-128	Reserved	Reserved
00 01DC 8000	00 01DC 807F	128	Tracer CFG25	Tracer CFG25
00 01DC 8080	00 01DC FFFF	32K-128	Reserved	Reserved
00 01DD 0000	00 01DD 007F	128	Tracer CFG26	Tracer CFG26
00 01DD 0080	00 01DD 7FFF	32K-128	Reserved	Reserved
00 01DD 8000	00 01DD 807F	128	Tracer CFG27	Tracer CFG27
00 01DD 8080	00 01DD FFFF	32K-128	Reserved	Reserved
00 01DE 0000	00 01DE 007F	128	Tracer CFG28	Tracer CFG28
00 01DE 0080	00 01DE 03FF	1K-128	Reserved	Reserved
00 01DE 0400	00 01DE 047F	128	Tracer CFG29	Tracer CFG29
00 01DD 0480	00 01DD 07FF	1K-128	Reserved	Reserved
00 01DE 0800	00 01DE 087F	128	Tracer CFG30	Tracer CFG30
00 01DE 0880	00 01DE 7FFF	30K-128	Reserved	Reserved
00 01DE 8000	00 01DE 807F	128	Tracer CFG31	Tracer CFG31
00 01DE 8080	00 01DF FFFF	64K-128	Reserved	Reserved
00 01E0 0000	00 01E3 FFFF	256K	Reserved	Reserved
00 01E4 0000	00 01E7FFFF	256k	TSIP_CFG	TSIP_CFG
00 01E8 0000	00 01E8 3FFF	16K	ARM CorePac_CFG	ARM CorePac_CFG
00 01E8 4000	00 01EB FFFF	240k	Reserved	Reserved
00 01EC 0000	00 01EF FFFF	256K	Reserved	Reserved

PRODUCT PREVIEW

Table 3-1 Device Memory Map Summary for AM5K2E04/02 (Part 3 of 12)

Physical 40 bit Address		Bytes	ARM View	SOC View
Start	End			
00 01F0 0000	00 01F7 FFFF	512K	Reserved	Reserved
00 01F8 0000	00 01F8 FFFF	64K	Reserved	Reserved
00 01F9 0000	00 01F9 FFFF	64K	Reserved	Reserved
00 01FA 0000	00 01FB FFFF	128K	Reserved	Reserved
00 01FC 0000	00 01FD FFFF	128K	Reserved	Reserved
00 01FE 0000	00 01FF FFFF	128K	Reserved	Reserved
00 0200 0000	00 020F FFFF	1M	Network Coprocessor 0(Packet Accelerator, 1-gigabit Ethernet switch subsystem and Security Accelerator)	Network Coprocessor 0(Packet Accelerator, 1-gigabit Ethernet switch subsystem and Security Accelerator)
00 0210 0000	00 0210 FFFF	64K	Reserved	Reserved
00 0211 0000	00 0211 FFFF	64K	Reserved	Reserved
00 0212 0000	00 0213 FFFF	128K	Reserved	Reserved
00 0214 0000	00 0215 FFFF	128K	Reserved	Reserved
00 0216 0000	00 0217 FFFF	128K	Reserved	Reserved
00 0218 0000	00 0218 7FFF	32k	Reserved	Reserved
00 0218 8000	00 0218 FFFF	32k	Reserved	Reserved
00 0219 0000	00 0219 FFFF	64k	Reserved	Reserved
00 021A 0000	00 021A FFFF	64K	Reserved	Reserved
00 021B 0000	00 021B FFFF	64K	Reserved	Reserved
00 021C 0000	00 021C 03FF	1K	Reserved	Reserved
00 021C 0400	00 021C 3FFF	15K	Reserved	Reserved
00 021C 4000	00 021C 43FF	1K	Reserved	Reserved
00 021C 4400	00 021C 5FFF	7K	Reserved	Reserved
00 021C 6000	00 021C 63FF	1K	Reserved	Reserved
00 021C 6400	00 021C 7FFF	7K	Reserved	Reserved
00 021C 8000	00 021C 83FF	1K	Reserved	Reserved
00 021C 8400	00 021C FFFF	31K	Reserved	Reserved
00 021D 0000	00 021D 03FF	1K	Memory protection unit (MPU) 15	Memory protection unit (MPU) 15
00 021D 0400	00 021D 047F	128	Tracer CFG32	Tracer CFG32
00 021D 0100	00 021D 3FFF	15K-128	Reserved	Reserved
00 021D 4000	00 021D 40FF	256	Reserved	Reserved
00 021D 4100	00 021D 7FFF	16K-256	Reserved	Reserved
00 021D 8000	00 021D 80FF	256	Reserved	Reserved
00 021D 8100	00 021D BFFF	16K-256	Reserved	Reserved
00 021D C000	00 021D C0FF	256	Reserved	Reserved
00 021D C100	00 021D EFFF	12K-256	Reserved	Reserved
00 021D F000	00 021D F07F	128	Reserved	Reserved
00 021D F080	00 021D FFFF	4K-128	Reserved	Reserved
00 021E 0000	00 021E FFFF	64K	Reserved	Reserved
00 021F 0000	00 021F 07FF	2K	Reserved	Reserved
00 021F 0800	00 021F 0FFF	2K	Reserved	Reserved
00 021F 1000	00 021F 17FF	2K	Reserved	Reserved
00 021F 1800	00 021F 3FFF	10K	Reserved	Reserved
00 021F 4000	00 021F 47FF	2K	Reserved	Reserved

PRODUCT PREVIEW

Table 3-1 Device Memory Map Summary for AM5K2E04/02 (Part 4 of 12)

Physical 40 bit Address		Bytes	ARM View	SOC View
Start	End			
00 021F 4800	00 021F 7FFF	14K	Reserved	Reserved
00 021F 8000	00 021F 87FF	2K	Reserved	Reserved
00 021F 8800	00 021F BFFF	14K	Reserved	Reserved
00 021F C000	00 021F C7FF	2K	Reserved	Reserved
00 021F C800	00 021F FFFF	14K	Reserved	Reserved
00 0220 0000	00 0220 007F	128	Reserved	Reserved
00 0220 0080	00 0220 FFFF	64K-128	Reserved	Reserved
00 0221 0000	00 0221 007F	128	Reserved	Reserved
00 0221 0080	00 0221 FFFF	64K-128	Reserved	Reserved
00 0222 0000	00 0222 007F	128	Reserved	Reserved
00 0222 0080	00 0222 FFFF	64K-128	Reserved	Reserved
00 0223 0000	00 0223 007F	128	Reserved	Reserved
00 0223 0080	00 0223 FFFF	64K-128	Reserved	Reserved
00 0224 0000	00 0224 007F	128	Reserved	Reserved
00 0224 0080	00 0224 FFFF	64K-128	Reserved	Reserved
00 0225 0000	00 0225 007F	128	Reserved	Reserved
00 0225 0080	00 0225 FFFF	64K-128	Reserved	Reserved
00 0226 0000	00 0226 007F	128	Reserved	Reserved
00 0226 0080	00 0226 FFFF	64K-128	Reserved	Reserved
00 0227 0000	00 0227 007F	128	Reserved	Reserved
00 0227 0080	00 0227 FFFF	64K-128	Reserved	Reserved
00 0228 0000	00 0228 007F	128	Timer 8	Timer 8
00 0228 0080	00 0228 FFFF	64K-128	Reserved	Reserved
00 0229 0000	00 0229 007F	128	Timer 9	Timer 9
00 0229 0080	00 0229 FFFF	64K-128	Reserved	Reserved
00 022A 0000	00 022A 007F	128	Timer 10	Timer 10
00 022A 0080	00 022A FFFF	64K-128	Reserved	Reserved
00 022B 0000	00 022B 007F	128	Timer 11	Timer 11
00 022B 0080	00 022B FFFF	64K-128	Reserved	Reserved
00 022C 0000	00 022C 007F	128	Timer 12	Timer 12
00 022C 0080	00 022C FFFF	64K-128	Reserved	Reserved
00 022D 0000	00 022D 007F	128	Timer 13	Timer 13
00 022D 0080	00 022D FFFF	64K-128	Reserved	Reserved
00 022E 0000	00 022E 007F	128	Timer 14	Timer 14
00 022E 0080	00 022E FFFF	64K-128	Reserved	Reserved
00 022F 0000	00 022F 007F	128	Timer 15	Timer 15
00 022F 0080	00 022F 00FF	128	Timer 16	Timer 16
00 022F 0100	00 022F 017F	128	Timer 17	Timer 17
00 022F 0180	00 022F 01FF	128	Timer 18	Timer 18
00 022F 0200	00 022F 027F	128	Timer 19	Timer 19
00 0230 0000	00 0230 FFFF	64K	Reserved	Reserved
00 0231 0000	00 0231 01FF	512	PLL Controller	PLL Controller
00 0231 0200	00 0231 9FFF	40K-512	Reserved	Reserved

PRODUCT PREVIEW

Table 3-1 Device Memory Map Summary for AM5K2E04/02 (Part 5 of 12)

Physical 40 bit Address		Bytes	ARM View	SOC View
Start	End			
00 0231 A000	00 0231 BFFF	8K	HyperLink0 SerDes Config	HyperLink0 SerDes Config
00 0231 C000	00 0231 DFFF	8K	Reserved	Reserved
00 0231 E000	00 0231 FFFF	8K	Reserved	Reserved
00 0232 0000	00 0232 3FFF	16K	PCIE 0 SerDes Config	PCIE 0 SerDes Config
00 0232 4000	00 0232 5FFF	8K	SGMII 1 SerDes Config	SGMII 1 SerDes Config
00 0232 5000	00 0232 7FFF	8K	PCIE 1SerDes Config	PCIE 1SerDes Config
00 0232 8000	00 0232 8FFF	4K	Reserved	Reserved
00 0232 9000	00 0232 9FFF	4K	DDRA PHY Config	DDRA PHY Config
00 0232 A000	00 0232 BFFF	8K	SGMII 0 SerDes Config	SGMII 0 SerDes Config
00 0232 C000	00 0232 CFFF	4K	SRIO SerDes Config	SRIO SerDes Config
00 0232 D000	00 0232 DFFF	4K	Reserved	Reserved
00 0232 E000	00 0232 EFFF	4K	Reserved	Reserved
00 0232 F000	00 0232 FFFF	4K	Reserved	Reserved
00 0233 0000	00 0233 03FF	1K	SmartReflex0	SmartReflex0
00 0233 0400	00 0233 07FF	1K	Reserved	Reserved
00 0233 0400	00 0233 FFFF	62K	Reserved	Reserved
00 0234 0000	00 0234 00FF	256	Reserved	Reserved
00 0234 0100	00 0234 3FFF	16K	Reserved	Reserved
00 0234 4000	00 0234 40FF	256	Reserved	Reserved
00 0234 4100	00 0234 7FFF	16K	Reserved	Reserved
00 0234 8000	00 0234 80FF	256	Reserved	Reserved
00 0234 8100	00 0234 BFFF	16K	Reserved	Reserved
00 0234 C000	00 0234 C0FF	256	Reserved	Reserved
00 0234 C100	00 0234 FFFF	16K	Reserved	Reserved
00 0235 0000	00 0235 0FFF	4K	Power sleep controller (PSC)	Power sleep controller (PSC)
00 0235 1000	00 0235 FFFF	64K-4K	Reserved	Reserved
00 0236 0000	00 0236 03FF	1K	Memory protection unit (MPU) 0	Memory protection unit (MPU) 0
00 0236 0400	00 0236 7FFF	31K	Reserved	Reserved
00 0236 8000	00 0236 83FF	1K	Memory protection unit (MPU) 1	Memory protection unit (MPU) 1
00 0236 8400	00 0236 FFFF	31K	Reserved	Reserved
00 0237 0000	00 0237 03FF	1K	Memory protection unit (MPU) 2	Memory protection unit (MPU) 2
00 0237 0400	00 0237 7FFF	31K	Reserved	Reserved
00 0237 8000	00 0237 83FF	1K	Reserved	Reserved
00 0237 8400	00 0237 FFFF	31K	Reserved	Reserved
00 0238 0000	00 0238 03FF	1K	Reserved	Reserved
00 0238 8000	00 0238 83FF	1K	Memory protection unit (MPU) 5	Memory protection unit (MPU) 5
00 0238 8400	00 0238 87FF	1K	Reserved	Reserved
00 0238 8800	00 0238 8BFF	1K	Memory protection unit (MPU) 7	Memory protection unit (MPU) 7
00 0238 8C00	00 0238 8FFF	1K	Memory protection unit (MPU) 8	Memory protection unit (MPU) 8
00 0238 9000	00 0238 93FF	1K	Memory protection unit (MPU) 9	Memory protection unit (MPU) 9
00 0238 9400	00 0238 97FF	1K	Memory protection unit (MPU) 10	Memory protection unit (MPU) 10
00 0238 9800	00 0238 9BFF	1K	Memory protection unit (MPU) 11	Memory protection unit (MPU) 11
00 0238 9C00	00 0238 9FFF	1K	Memory protection unit (MPU) 12	Memory protection unit (MPU) 12

Table 3-1 Device Memory Map Summary for AM5K2E04/02 (Part 6 of 12)

Physical 40 bit Address		Bytes	ARM View	SOC View
Start	End			
00 0238 A000	00 0238 A3FF	1K	Memory protection unit (MPU) 13	Memory protection unit (MPU) 13
00 0238 A400	00 0238 A7FF	1K	Memory protection unit (MPU) 14	Memory protection unit (MPU) 14
00 0238 A800	00 023F FFFF	471K	Reserved	Reserved
00 0240 0000	00 0243 FFFF	256K	Reserved	Reserved
00 0244 0000	00 0244 3FFF	16K	Reserved	Reserved
00 0244 4000	00 0244 FFFF	48K	Reserved	Reserved
00 0245 0000	00 0245 3FFF	16K	Reserved	Reserved
00 0245 4000	00 0245 FFFF	48K	Reserved	Reserved
00 0246 0000	00 0246 3FFF	16K	Reserved	Reserved
00 0246 4000	00 0246 FFFF	48K	Reserved	Reserved
00 0247 0000	00 0247 3FFF	16K	Reserved	Reserved
00 0247 4000	00 0247 FFFF	48K	Reserved	Reserved
00 0248 0000	00 0248 3FFF	16K	Reserved	Reserved
00 0248 4000	00 0248 FFFF	48K	Reserved	Reserved
00 0249 0000	00 0249 3FFF	16K	Reserved	Reserved
00 0249 4000	00 0249 FFFF	48K	Reserved	Reserved
00 024A 0000	00 024A 3FFF	16K	Reserved	Reserved
00 024A 4000	00 024A FFFF	48K	Reserved	Reserved
00 024B 0000	00 024B 3FFF	16K	Reserved	Reserved
00 024B 4000	00 024B FFFF	48K	Reserved	Reserved
00 024C 0000	00 024C 01FF	512	Reserved	Reserved
00 024C 0200	00 024C 03FF	1K-512	Reserved	Reserved
00 024C 0400	00 024C 07FF	1K	Reserved	Reserved
00 024C 0800	00 024C FFFF	62K	Reserved	Reserved
00 024D 0000	00 024F FFFF	192K	Reserved	Reserved
00 0250 0000	00 0250 007F	128	Reserved	Reserved
00 0250 0080	00 0250 7FFF	32K-128	Reserved	Reserved
00 0250 8000	00 0250 FFFF	32K	Reserved	Reserved
00 0251 0000	00 0251 FFFF	64K	Reserved	Reserved
00 0252 0000	00 0252 03FF	1K	Reserved	Reserved
00 0252 0400	00 0252 FFFF	64K-1K	Reserved	Reserved
00 0253 0000	00 0253 007F	128	I^2C0	I^2C0
00 0253 0080	00 0253 03FF	1K-128	Reserved	Reserved
00 0253 0400	00 0253 047F	128	I^2C1	I^2C1
00 0253 0480	00 0253 07FF	1K-128	Reserved	Reserved
00 0253 0800	00 0253 087F	128	I^2C2	I^2C2
00 0253 0880	00 0253 0BFF	1K-128	Reserved	Reserved
00 0253 0C00	00 0253 0C3F	64	UART0	UART0
00 0253 0C40	00 0253 FFFF	1K-64	Reserved	Reserved
00 0253 1000	00 0253 103F	64	UART1	UART1
00 0253 1040	00 0253 FFFF	60K-64	Reserved	Reserved
00 0254 0000	00 0255 FFFF	128K	Reserved	Reserved
00 0256 0080	00 0257 FFFF	128K	ARM CorePac INTC	ARM CorePac INTC

PRODUCT PREVIEW

Table 3-1 Device Memory Map Summary for AM5K2E04/02 (Part 7 of 12)

Physical 40 bit Address		Bytes	ARM View	SOC View
Start	End			
00 0258 0000	00 025F FFFF	512K	Reserved	Reserved
00 0260 0000	00 0260 1FFF	8K	Secondary interrupt controller (INTC) 0	Secondary interrupt controller (INTC) 0
00 0260 2000	00 0260 3FFF	8K	Reserved	Reserved
00 0260 4000	00 0260 5FFF	8K	Reserved	Reserved
00 0260 6000	00 0260 7FFF	8K	Reserved	Reserved
00 0260 8000	00 0260 9FFF	8K	Secondary interrupt controller (INTC) 2	Secondary interrupt controller (INTC) 2
00 0260 A000	00 0260 BEFF	8K-256	Reserved	Reserved
00 0260 BF00	00 0260 BFFF	256	GPIO Config	GPIO Config
00 0260 C000	00 0261 BFFF	64K	Reserved	Reserved
00 0261 C000	00 0261 FFFF	16K	Reserved	Reserved
00 0262 0000	00 0262 0FFF	4K	BOOTCFG chip-level registers	BOOTCFG chip-level registers
00 0262 1000	00 0262 FFFF	60K	Reserved	Reserved
00 0263 0000	00 0263 FFFF	64K	USB 0 PHY CFG	USB 0 PHY CFG
00 0264 0000	00 0264 07FF	2K	Semaphore Config	Semaphore Config
00 0264 0800	00 0264 FFFF	62K	Reserved	Reserved
00 0265 0000	00 0267 FFFF	192K	Reserved	Reserved
00 0268 0000	00 0268 FFFF	512K	USB 0 MMR CFG	USB 0 MMR CFG
00 0270 0000	00 0270 7FFF	32K	EDMA channel controller (TPCC) 0	EDMA channel controller (TPCC) 0
00 0270 8000	00 0270 FFFF	32K	EDMA channel controller (TPCC) 4	EDMA channel controller (TPCC) 4
00 0271 0000	00 0271 FFFF	64K	Reserved	Reserved
00 0272 0000	00 0272 7FFF	32K	EDMA channel controller (TPCC) 1	EDMA channel controller (TPCC) 1
00 0272 8000	00 0272 FFFF	32K	EDMA channel controller (TPCC) 3	EDMA channel controller (TPCC) 3
00 0273 0000	00 0273 FFFF	64K	Reserved	Reserved
00 0274 0000	00 0274 7FFF	32K	EDMA channel controller (TPCC) 2	EDMA channel controller (TPCC) 2
00 0274 8000	00 0275 FFFF	96K	Reserved	Reserved
00 0276 0000	00 0276 03FF	1K	EDMA TPCC0 transfer controller (TPTC) 0	EDMA TPCC0 transfer controller (TPTC) 0
00 0276 0400	00 0276 7FFF	31K	Reserved	Reserved
00 0276 8000	00 0276 83FF	1K	EDMA TPCC0 transfer controller (TPTC) 1	EDMA TPCC0 transfer controller (TPTC) 1
00 0276 8400	00 0276 FFFF	31K	Reserved	Reserved
00 0277 0000	00 0277 03FF	1K	EDMA TPCC1 transfer controller (TPTC) 0	EDMA TPCC1 transfer controller (TPTC) 0
00 0277 0400	00 0277 7FFF	31K	Reserved	Reserved
00 0277 8000	00 0277 83FF	1K	EDMA TPCC1 transfer controller (TPTC) 1	EDMA TPCC1 transfer controller (TPTC) 1
00 0278 0400	00 0277 FFFF	31K	Reserved	Reserved
00 0278 0000	00 0278 03FF	1K	EDMA TPCC1 transfer controller (TPTC) 2	EDMA TPCC1 transfer controller (TPTC) 2
00 0278 0400	00 0278 7FFF	31K	Reserved	Reserved
00 0278 8000	00 0278 83FF	1K	EDMA TPCC1 transfer controller (TPTC) 3	EDMA TPCC1 transfer controller (TPTC) 3
00 0278 8400	00 0278 FFFF	31K	Reserved	Reserved
00 0279 0000	00 0279 03FF	1K	EDMA TPCC2 transfer controller (TPTC) 0	EDMA TPCC2 transfer controller (TPTC) 0
00 0279 0400	00 0279 7FFF	31K	Reserved	Reserved
00 0279 8000	00 0279 83FF	1K	EDMA TPCC2 transfer controller (TPTC) 1	EDMA TPCC2 transfer controller (TPTC) 1
00 0279 8400	00 0279 FFFF	31K	Reserved	Reserved
00 027A 0000	00 027A 03FF	1K	EDMA TPCC2 transfer controller (TPTC) 2	EDMA TPCC2 transfer controller (TPTC) 2
00 027A 0400	00 027A 7FFF	31K	Reserved	Reserved

Table 3-1 Device Memory Map Summary for AM5K2E04/02 (Part 8 of 12)

Physical 40 bit Address		Bytes	ARM View	SOC View
Start	End			
00 027A 8000	00 027A 83FF	1K	EDMA TPCC2 transfer controller (TPTC) 3	EDMA TPCC2 transfer controller (TPTC) 3
00 027A 8400	00 027A FFFF	31K	Reserved	Reserved
00 027B 0000	00 027B 03FF	1K	EDMA TPCC3 transfer controller (TPTC) 0	EDMA TPCC3 transfer controller (TPTC) 0
00 027B 0400	00 027B 7FFF	31K	Reserved	Reserved
00 027B 8000	00 027B 83FF	1K	EDMA TPCC3 transfer controller (TPTC) 1	EDMA TPCC3 transfer controller (TPTC) 1
00 027B 8400	00 027B 87FF	1K	EDMA TPCC4 transfer controller (TPTC) 0	EDMA TPCC4 transfer controller (TPTC) 0
00 027B 8800	00 027B 8BFF	1K	EEDMA TPCC4 transfer controller (TPTC) 1	EEDMA TPCC4 transfer controller (TPTC) 1
00 027B 8C00	00 027B FFFF	29K	Reserved	Reserved
00 027C 0000	00 027C 03FF	1K	Reserved	Reserved
00 027C 0400	00 027C FFFF	63K	Reserved	Reserved
00 027D 0000	00 027D 3FFF	16K	TI embedded trace buffer (TETB) - CorePac0	TI embedded trace buffer (TETB) - CorePac0
00 027D 4000	00 027D 7FFF	16K	TBR_ARM CorePac - Trace buffer - ARM CorePac	TBR_ARM CorePac - Trace buffer - ARM CorePac
00 027D 8000	00 027D FFFF	32K	Reserved	Reserved
00 027E 0000	00 027E 3FFF	16K	Reserved	Reserved
00 027E 4000	00 027E FFFF	48K	Reserved	Reserved
00 027F 0000	00 027F 3FFF	16K	Reserved	Reserved
00 027F 4000	00 027F FFFF	48K	Reserved	Reserved
00 0280 0000	00 0280 3FFF	16K	Reserved	Reserved
00 0280 4000	00 0280 FFFF	48K	Reserved	Reserved
00 0281 0000	00 0281 3FFF	16K	Reserved	Reserved
00 0281 4000	00 0281 FFFF	48K	Reserved	Reserved
00 0282 0000	00 0282 3FFF	16K	Reserved	Reserved
00 0282 4000	00 0282 FFFF	48K	Reserved	Reserved
00 0283 0000	00 0283 3FFF	16K	Reserved	Reserved
00 0283 4000	00 0283 FFFF	48K	Reserved	Reserved
00 0284 0000	00 0284 3FFF	16K	Reserved	Reserved
00 0284 4000	00 0284 FFFF	48K	Reserved	Reserved
00 0285 0000	00 0285 7FFF	32K	TBR_SYS- Trace buffer - System	TBR_SYS- Trace buffer - System
00 0285 8000	00 0285 FFFF	32K	Reserved	Reserved
00 0286 0000	00 028F FFFF	640K	Reserved	Reserved
00 0290 0000	00 0293 FFFF	256K	Reserved	Reserved
00 0294 0000	00 029F FFFF	768K	Reserved	Reserved
00 02A0 0000	00 02AF FFFF	1M	Navigator configuration	Navigator configuration
00 02B0 0000	00 02BF FFFF	1M	Navigator linking RAM	Navigator linking RAM
00 02C0 0000	00 02C0 FFFF	64K	Reserved	Reserved
00 02C1 0000	00 02C1 FFFF	64K	Reserved	Reserved
00 02C2 0000	00 02C3 FFFF	128K	Reserved	Reserved
00 02C4 0000	00 02C5 FFFF	128K	Reserved	Reserved
00 02C6 0000	00 02C7 FFFF	128K	Reserved	Reserved
00 02C8 0000	00 02C8 FFFF	64K	Reserved	Reserved
00 02C9 0000	00 02C9 FFFF	64K	Reserved	Reserved
00 02CA 0000	00 02CB FFFF	128K	Reserved	Reserved

Table 3-1 Device Memory Map Summary for AM5K2E04/02 (Part 9 of 12)

Physical 40 bit Address		Bytes	ARM View	SOC View
Start	End			
00 02CC 0000	00 02CD FFFF	128K	Reserved	Reserved
00 02CE 0000	00 02EF FFFF	15M-896K	Reserved	Reserved
00 02F0 0000	00 02FF FFFF	1M	10GbE Config	10GbE Config
00 0300 0000	00 030F FFFF	1M	DBG Config	DBG Config
00 0310 0000	00 07FF FFFF	79M	Reserved	Reserved
00 0800 0000	00 0801 FFFF	128K	Extended memory controller (XMC) configuration	Extended memory controller (XMC) configuration
00 0802 0000	00 0BBF FFFF	60M-128K	Reserved	Reserved
00 0BC0 0000	00 0BCF FFFF	1M	Multicore shared memory controller (MSMC) config	Multicore shared memory controller (MSMC) config
00 0BD0 0000	00 0BFF FFFF	3M	Reserved	Reserved
00 0C00 0000	00 0C1F FFFF	2M	Multicore shared memory (MSM)	Multicore shared memory (MSM)
00 0C20 0000	00 0C5F FFFF	4M	Reserved	Reserved
00 0C60 0000	00 0FFF FFFF	58M	Reserved	Reserved
00 1000 0000	00 107F FFFF	8M	Reserved	Reserved
00 1080 0000	00 108F FFFF	1M	Reserved	Reserved
00 1090 0000	00 10DF FFFF	5M	Reserved	Reserved
00 10E0 0000	00 10E0 7FFF	32K	Reserved	Reserved
00 10E0 8000	00 10EF FFFF	1M-32K	Reserved	Reserved
00 10F0 0000	00 10F0 7FFF	32K	Reserved	Reserved
00 10F0 8000	00 117F FFFF	9M-32K	Reserved	Reserved
00 1180 0000	00 118F FFFF	1M	Reserved	Reserved
00 1190 0000	00 11DF FFFF	5M	Reserved	Reserved
00 11E0 0000	00 11E0 7FFF	32K	Reserved	Reserved
00 11E0 8000	00 11EF FFFF	1M-32K	Reserved	Reserved
00 11F0 0000	00 11F0 7FFF	32K	Reserved	Reserved
00 11F0 8000	00 127F FFFF	9M-32K	Reserved	Reserved
00 1280 0000	00 128F FFFF	1M	Reserved	Reserved
00 1290 0000	00 12DF FFFF	5M	Reserved	Reserved
00 12E0 0000	00 12E0 7FFF	32K	Reserved	Reserved
00 12E0 8000	00 12EF FFFF	1M-32K	Reserved	Reserved
00 12F0 0000	00 12F0 7FFF	32K	Reserved	Reserved
00 12F0 8000	00 137F FFFF	9M-32K	Reserved	Reserved
00 1380 0000	00 1388 FFFF	1M	Reserved	Reserved
00 1390 0000	00 13DF FFFF	5M	Reserved	Reserved
00 13E0 0000	00 13E0 7FFF	32K	Reserved	Reserved
00 13E0 8000	00 13EF FFFF	1M-32K	Reserved	Reserved
00 13F0 0000	00 13F0 7FFF	32K	Reserved	Reserved
00 13F0 8000	00 147F FFFF	9M-32K	Reserved	Reserved
00 1480 0000	00 148F FFFF	1M	Reserved	Reserved
00 1490 0000	00 14DF FFFF	5M	Reserved	Reserved
00 14E0 0000	00 14E0 7FFF	32K	Reserved	Reserved
00 14E0 8000	00 14EF FFFF	1M-32K	Reserved	Reserved
00 14F0 0000	00 14F0 7FFF	32K	Reserved	Reserved

Table 3-1 Device Memory Map Summary for AM5K2E04/02 (Part 10 of 12)

Physical 40 bit Address		Bytes	ARM View	SOC View
Start	End			
00 14F0 8000	00 157F FFFF	9M-32K	Reserved	Reserved
00 1580 0000	00 158F FFFF	1M	Reserved	Reserved
00 1590 0000	00 15DF FFFF	5M	Reserved	Reserved
00 15E0 0000	00 15E0 7FFF	32K	Reserved	Reserved
00 15E0 8000	00 15EF FFFF	1M-32K	Reserved	Reserved
00 15F0 0000	00 15F0 7FFF	32K	Reserved	Reserved
00 15F0 8000	00 167F FFFF	9M-32K	Reserved	Reserved
00 1680 0000	00 168F FFFF	1M	Reserved	Reserved
00 1690 0000	00 16DF FFFF	5M	Reserved	Reserved
00 16E0 0000	00 16E0 7FFF	32K	Reserved	Reserved
00 16E0 8000	00 16EF FFFF	1M-32K	Reserved	Reserved
00 16F0 0000	00 16F0 7FFF	32K	Reserved	Reserved
00 16F0 8000	00 177F FFFF	9M-32K	Reserved	Reserved
00 1780 0000	00 178F FFFF	1M	Reserved	Reserved
00 1790 0000	00 17DF FFFF	5M	Reserved	Reserved
00 17E0 0000	00 17E0 7FFF	32K	Reserved	Reserved
00 17E0 8000	00 17EF FFFF	1M-32K	Reserved	Reserved
00 17F0 0000	00 17F0 7FFF	32K	Reserved	Reserved
00 17F0 8000	00 1FFF FFFF	129M-32K	Reserved	Reserved
00 2000 0000	00 200F FFFF	1M	System trace manager (STM) configuration	System trace manager (STM) configuration
00 2010 0000	00 201F FFFF	1M	Reserved	Reserved
00 2020 0000	00 205F FFFF	4M	Reserved	Reserved
00 2060 0000	00 206F FFFF	1M	Network Coprocessor 1(Packet Accelerator, 1-gigabit Ethernet switch subsystem and Security Accelerator)	Network Coprocessor 1(Packet Accelerator, 1-gigabit Ethernet switch subsystem and Security Accelerator)
00 2070 0000	00 2077 FFFF	512K	USB 1 MMR CFG	USB 1 MMR CFG
00 2078 0000	00 2078 FFFF	64K	USB 1 PHY CFG	USB 1 PHY CFG
00 2079 0000	00 207F FFFF	448K	Reserved	Reserved
00 2080 0000	00 208F FFFF	1M	Reserved	Reserved
00 2090 0000	00 209F FFFF	1M	Reserved	Reserved
00 20A0 0000	00 20A3 FFFF	256K	Reserved	Reserved
00 20A4 0000	00 20A4 FFFF	64K	Reserved	Reserved
00 20A5 0000	00 20AF FFFF	704K	Reserved	Reserved
00 20B0 0000	00 20B3 FFFF	256K	Boot ROM	Boot ROM
00 20B4 0000	00 20BE FFFF	704K	Reserved	Reserved
00 20BF 0000	00 20BF 01FF	64K	Reserved	Reserved
00 20C0 0000	00 20FF FFFF	4M	Reserved	Reserved
00 2100 0000	00 2100 03FF	1K	Reserved	Reserved
00 2100 0400	00 2100 05FF	512	SPI0	SPI0
00 2100 0600	00 2100 07FF	512	SPI1	SPI1
00 2100 0800	00 2100 09FF	512	SPI2	SPI2
00 2100 0A00	00 2100 0AFF	256	EMIF Config	EMIF Config
00 2100 0B00	00 2100 FFFF	62K-768	Reserved	Reserved
00 2101 0000	00 2101 01FF	512	DDR3A EMIF Config	DDR3A EMIF Config

PRODUCT PREVIEW

Table 3-1 Device Memory Map Summary for AM5K2E04/02 (Part 11 of 12)

Physical 40 bit Address		Bytes	ARM View	SOC View
Start	End			
00 2101 0200	00 2101 07FF	2K-512	Reserved	Reserved
00 2101 0800	00 2101 09FF	512	Reserved	Reserved
00 2101 0A00	00 2101 0FFF	2K-512	Reserved	Reserved
00 2101 1000	00 2101 FFFF	60K	Reserved	Reserved
00 2102 0000	00 2102 7FFF	32K	PCIe 1config	PCIe 1config
00 2102 8000	00 2103 FFFF	96K	Reserved	Reserved
00 2104 0000	00 217F FFFF	4M-256K	Reserved	Reserved
00 2140 0000	00 2140 00FF	256	HyperLink0 config	HyperLink0 config
00 2140 0100	00 2140 01FF	256	Reserved	Reserved
00 2140 0400	00 217F FFFF	4M-512	Reserved	Reserved
00 2180 0000	00 2180 7FFF	32K	PCIe 0 config	PCIe 0 config
00 2180 8000	00 21BF FFFF	4M-32K	Reserved	Reserved
00 21C0 0000	00 21FF FFFF	4M	Reserved	Reserved
00 2200 0000	00 229F FFFF	10M	Reserved	Reserved
00 22A0 0000	00 22A0 FFFF	64K	Reserved	Reserved
00 22A1 0000	00 22AF FFFF	1M-64K	Reserved	Reserved
00 22B0 0000	00 22B0 FFFF	64K	Reserved	Reserved
00 22B1 0000	00 22BF FFFF	1M-64K	Reserved	Reserved
00 22C0 0000	00 22C0 FFFF	64K	Reserved	Reserved
00 22C1 0000	00 22CF FFFF	1M-64K	Reserved	Reserved
00 22D0 0000	00 22D0 FFFF	64K	Reserved	Reserved
00 22D1 0000	00 22DF FFFF	1M-64K	Reserved	Reserved
00 22E0 0000	00 22E0 FFFF	64K	Reserved	Reserved
00 22E1 0000	00 22EF FFFF	1M-64K	Reserved	Reserved
00 22F0 0000	00 22F0 FFFF	64K	Reserved	Reserved
00 22F1 0000	00 22FF FFFF	1M-64K	Reserved	Reserved
00 2300 0000	00 2300 FFFF	64K	Reserved	Reserved
00 2301 0000	00 230F FFFF	1M-64K	Reserved	Reserved
00 2310 0000	00 2310 FFFF	64K	Reserved	Reserved
00 2311 0000	00 231F FFFF	1M-64K	Reserved	Reserved
00 2320 0000	00 2324 FFFF	384K	Reserved	Reserved
00 2325 0000	00 239F FFFF	8M-384K	Reserved	Reserved
00 23A0 0000	00 23BF FFFF	2M	Navigator	Navigator
00 23C0 0000	00 23FF FFFF	4M	Reserved	Reserved
00 2400 0000	00 27FF FFFF	64M	Reserved	Reserved
00 2800 0000	00 2FFF FFFF	128M	Reserved	Reserved
00 3000 0000	00 33FF FFFF	64M	EMIF16 CS2	EMIF16 CS2
00 3400 0000	00 37FF FFFF	64M	EMIF16 CS3	EMIF16 CS3
00 3800 0000	00 3BFF FFFF	64M	EMIF16 CS4	EMIF16 CS4
00 3C00 0000	00 3FFF FFFF	64M	EMIF16 CS5	EMIF16 CS5
00 4000 0000	00 4FFF FFFF	256M	HyperLink0 data	HyperLink0 data
00 5000 0000	00 5FFF FFFF	256M	PCIe 0 data	PCIe 0 data
00 6000 0000	00 6FFF FFFF	256M	PCIe 1data	PCIe 1data

Table 3-1 Device Memory Map Summary for AM5K2E04/02 (Part 12 of 12)

Physical 40 bit Address		Bytes	ARM View	SOC View
Start	**End**			
00 7000 0000	00 FFFF FFFF	2304M	Reserved	Reserved
01 0000 0000	01 20FF FFFF	528M	Reserved	Reserved
01 2100 0000	01 2100 01FF	512	Reserved	DDR3A EMIF configuration
01 2100 0200	07 FFFF FFFF	32G-512	Reserved	Reserved
08 0000 0000	09 FFFF FFFF	8G	DDR3A data	DDR3A data
0A 0000 0000	FF FFFF FFFF	984G	Reserved	Reserved
End of Table 3-1				

PRODUCT PREVIEW

3.2 Memory Protection Unit (MPU) for AM5K2E04/02

CFG (configuration) space of all slave devices on the TeraNet is protected by the MPU. The AM5K2E04/02 contains sixteen MPUs of which thirteen MPUs are used:

- MPU0 is used to protect main CORE/3 CFG TeraNet_3P_B (SCR_3P (B)) .
- MPU1/2/5 are used for QM_SS (one for VBUSM port and one each for the two configuration VBUSP port).
- MPU3/4/6 are not used
- MPU7 is used for PCIE1
- MPU8 is used for peripherals connected to TeraNet_6P_A (SCR_6P (A)).
- MPU9 is used for interrupt controllers connected to TeraNet_3P (SCR_3P).
- MPU10 is used for semaphore.
- MPU11 is used to protect TeraNet_6P_B (SCR_6P (B)) CPU/6 CFG TeraNet
- MPU12/13/14 are used for SPI0/1/2
- MPU15 is used for NETCP1/USB1

This section contains MPU register map and details of device-specific MPU registers only. For MPU features and details of generic MPU registers, see the *Memory Protection Unit (MPU) for KeyStone Devices User Guide* in 1.8 "Related Documentation from Texas Instruments" on page 13.

The following tables show the configuration of each MPU and the memory regions protected by each MPU.

Table 3-2 MPU 0- MPU5 Default Configuration

Setting	MPU0 Main SCR_3P (B)	MPU1 (QM_SS DATA PORT)	MPU2 (QM_SS CFG1 PORT)	MPU3	MPU4	MPU5 (QM_SS CFG1 PORT)
Default permission	Assume allowed	Assume allowed	Assume allowed	Reserved	Reserved	Assume allowed
Number of allowed IDs supported	16	16	16			16
Number of programmable ranges supported	16	16	16			16
Compare width	1KB granularity	1KB granularity	1KB granularity			1KB granularity
End of Table 3-2						

Table 3-3 MPU 6-MPU11 Default Configuration

Setting	MPU6	MPU7 PCIE1	MPU8 EMIF16	MPU9 INTC	MPU10 SM	MPU11 SCR_6P (B)
Default permission	Reserved	Assume allowed	Assume allowed	Assume allowed	Assume allowed	Assume allowed
Number of allowed IDs supported		16	16	16	16	16
Number of programmable ranges supported		16	8	4	2	16
Compare width		1KB granularity	1KB granularity	1KB granularity	1KB granularity	1KB granularity
End of Table 3-3						

Table 3-4 MPU12- MPU15 Default Configuration

Setting	MPU12 SPI0	MPU13 SPI1	MPU14 SPI2	MPU15 NETCP1_USB1
Default permission	Assume allowed	Assume allowed	Assume allowed	Assume allowed
Number of allowed IDs supported	16	16	16	16
Number of programmable ranges supported	2	2	2	8
Compare width	1KB granularity	1KB granularity	1KB granularity	1KB granularity
End of Table 3-4				

Table 3-5 MPU Memory Regions

	Memory Protection	Start Address	End Address
MPU0	Main CFG SCR	0x01D0_0000	0x01e7_FFFF
MPU1	QM_SS DATA PORT	0x23A0_0000	0x23BF_FFFF
MPU2	QM_SS CFG1 PORT	0x02A0_0000	0x02AF_FFFF
MPU3	Reserved	N/A	N/A
MPU4	Reserved	N/A	N/A
MPU5	QM_SS CFG2 PORT	0x02A0_4000	0x02BF_FFFF
MPU6	Reserved	N/A	N/A
MPU7	PCIE1	0x2101_0000	0xFFFF_FFFF
MPU8	SPIROM/EMIF16	0x20B0_0000	0x3FFF_FFFF
MPU9	INTC/AINTC	0x0264_0000	0x0264_07FF
MPU10	Semaphore	0x0260_0000	0x0260_9FFF
MPU11	SCR_6 and CPU/6 CFG SCR	0x0220_0000	0x03FF_FFFF
MPU12	SPI0	0x2100_0400	0x2100_07FF
MPU13	SPI1	0x2100_0400	0x2100_07FF
MPU14	SPI2	0x2100_0800	0x2100_0AFF
MPU15	NETCP1_USB1	0x2060_0000	0x209F_FFFF
End of Table 3-5			

Table 3-6 shows the unique Master ID assigned to each CorePac and peripherals on the device.

Table 3-6 Master ID Settings (Part 1 of 4)

Master ID	AM5K2E04/02
0	Reserved
1	Reserved
2	Reserved
3	Reserved
4	Reserved
5	Reserved
6	Reserved
7	Reserved
8	ARM CorePac 0
9	ARM CorePac1
10	ARM CorePac 2
11	ARM CorePac 3
12	Reserved
13	Reserved

PRODUCT PREVIEW

Table 3-6 Master ID Settings (Part 2 of 4)

Master ID	AM5K2E04/02
14	Reserved
15	Reserved
16	Reserved
17	Reserved
18	Reserved
19	Reserved
20	Reserved
21	Reserved
22	Reserved
23	Reserved
24	Reserved
25	EDMA0_TC0 read
26	EDMA0_TC0 write
27	EDMA0_TC1 read
28	Hyperlink0
29	USB1
30	Reserved
31	PCIE0
32	EDMA0_TC1 write
33	EDMA1_TC0 read
34	EDMA1_TC0 write
35	EDMA1_TC1 read
36	EDMA1_TC1write
37	EDMA1_TC2 read
38	EDMA1_TC2 write
39	EDMA1_TC3 read
40	EDMA1_TC3 write
41	EDMA2_TC0 read
42	EDMA2_TC0 write
43	EDMA2_TC1 read
44	EDMA2_TC1 write
45	EDMA2_TC2 read
46	EDMA2_TC2 write
47	EDMA2_TC3 read
48	EDMA2_TC3 write
49	EDMA3_TC0 read
50	EDMA3_TC0 write
51	EDMA3_TC1 read
52	Reserved
53	EDMA3_TC1 write
54 to 55	Reserved
56	USB0
57	Reserved
58	Reserved

Table 3-6 Master ID Settings (Part 3 of 4)

Master ID	AM5K2E04/02
59	Reserved
60	Reserved
61	Reserved
62	EDMA3CC0
63	EDMA3CC1
64	EDMA3CC2
65	Reserved
66	Reserved
67	Reserved
68 to 71	Queue Manager
72 to 75	NETCP1
76 to 79	Reserved
80	TSIP
81	Reserved
82	Reserved
83	Reserved
84 to 87	Reserved
88 to 91	Reserved
92 to 95	Reserved
96 to 99	Packet DMA MST1
100 to 101	Reserved
102	PCIE1
103	Reserved
104	Reserved
105	Reserved
106	Reserved
107	DBG_DAP
108-139	Reserved
140	Reserved
141	Reserved
142	Reserved
143	Reserved
144	Reserved
145	Reserved
146	Reserved
147	Reserved
148	CPT_MSMC0
149	CPT_MSMC1
150	CPT_MSMC2
151	CPT_MSMC3
152	CPT_DDR3A
153	CPT_SM
154	CPT_QM_CFG1
155	CPT_QM_M

Table 3-6 Master ID Settings (Part 4 of 4)

Master ID	AM5K2E04/02
156	CPT_CFG
157	Reserved
158	Reserved
159	Reserved
160	CPT_QM_CFG2
161	CPT_PCIE1
162	Reserved
163	Reserved
164	CPT_EDMA3CC0_4
165	CPT_EDMA3CC1_2_3
166	CPT_INTC
167	CPT_SPI_ROM_EMIP16
168	Reserved
169	EDMA4_TC0 read
170	EDMA4_TC0 write
171	EDMA4_TC1 read
172	EDMA4_TC1 write
173	EDMA4_CC_TR
174	CPT_MSMC7
175	CPT_MSMC6
176	CPT_MSMC5
177	CPT_MSMC4
178	Reserved
179	Reserved
180-183	NETCP0
184-255	Reserved
End of Table 3-6	

Table 3-7 shows the privilege ID of each C66x CorePac and every mastering peripheral. The table also shows the privilege level (supervisor vs. user), security level (secure vs. non-secure), and access type (instruction read vs. data/DMA read or write) of each master on the device. In some cases, a particular setting depends on software being executed at the time of the access or the configuration of the master peripheral.

Table 3-7 Privilege ID Settings (Part 1 of 2)

Privilege ID	Master	Privilege Level	Access Type
0	C66x CorePac0	User/Supervisor (S/W dependant)	Instruction/Data
1	Reserved	N/A	N/A
2	Reserved	N/A	N/A
3	Reserved	N/A	N/A
4	Reserved	N/A	N/A
5	Reserved	N/A	N/A
6	Reserved	N/A	N/A
7	Reserved	N/A	N/A
8	ARM CorePac	User/Supervisor (S/W dependent)	Instruction/Data

Table 3-7 Privilege ID Settings (Part 2 of 2)

Privilege ID	Master	Privilege Level	Access Type
9	All packet DMA masters (Both NetCP, Both QM_CDMA) Both USB	User	Data
10	QM_SECOND	User	Data
11	PCIE0	User/Supervisor	Data
12	DAP	User/Supervisor (Emulation S/W dependent)	Data
13	PCIE1	User/Supervisor	Data
14	Hyperlink	User/Supervisor	Data
15	TSIP	User	Data
End of Table 3-7			

3.2.1 MPU Registers

This section includes the offsets for MPU registers and definitions for device-specific MPU registers. For Number of Programmable Ranges supported (PROGx_MPSA, PROGxMPEA) refer to the following tables.

3.2.1.1 MPU Register Map

Table 3-8 MPU Registers (Part 1 of 2)

Offset	Name	Description
0h	REVID	Revision ID
4h	CONFIG	Configuration
10h	IRAWSTAT	Interrupt raw status/set
14h	IENSTAT	Interrupt enable status/clear
18h	IENSET	Interrupt enable
1Ch	IENCLR	Interrupt enable clear
20h	EOI	End of interrupt
200h	PROG0_MPSAR	Programmable range 0, start address
204h	PROG0_MPEAR	Programmable range 0, end address
208h	PROG0_MPPAR	Programmable range 0, memory page protection attributes
210h	PROG1_MPSAR	Programmable range 1, start address
214h	PROG1_MPEAR	Programmable range 1, end address
218h	PROG1_MPPAR	Programmable range 1, memory page protection attributes
220h	PROG2_MPSAR	Programmable range 2, start address
224h	PROG2_MPEAR	Programmable range 2, end address
228h	PROG2_MPPAR	Programmable range 2, memory page protection attributes
230h	PROG3_MPSAR	Programmable range 3, start address
234h	PROG3_MPEAR	Programmable range 3, end address
238h	PROG3_MPPAR	Programmable range 3, memory page protection attributes
240h	PROG4_MPSAR	Programmable range 4, start address
244h	PROG4_MPEAR	Programmable range 4, end address
248h	PROG4_MPPAR	Programmable range 4, memory page protection attributes
250h	PROG5_MPSAR	Programmable range 5, start address
254h	PROG5_MPEAR	Programmable range 5, end address
258h	PROG5_MPPAR	Programmable range 5, memory page protection attributes
260h	PROG6_MPSAR	Programmable range 6, start address

PRODUCT PREVIEW

Table 3-8 MPU Registers (Part 2 of 2)

Offset	Name	Description
264h	PROG6_MPEAR	Programmable range 6, end address
268h	PROG6_MPPAR	Programmable range 6, memory page protection attributes
270h	PROG7_MPSAR	Programmable range 7, start address
274h	PROG7_MPEAR	Programmable range 7, end address
278h	PROG7_MPPAR	Programmable range 7, memory page protection attributes
280h	PROG8_MPSAR	Programmable range 8, start address
284h	PROG8_MPEAR	Programmable range 8, end address
288h	PROG8_MPPAR	Programmable range 8, memory page protection attributes
290h	PROG9_MPSAR	Programmable range 9, start address
294h	PROG9_MPEAR	Programmable range 9, end address
298h	PROG9_MPPAR	Programmable range 9, memory page protection attributes
2A0h	PROG10_MPSAR	Programmable range 10, start address
2A4h	PROG10_MPEAR	Programmable range 10, end address
2A8h	PROG10_MPPAR	Programmable range 10, memory page protection attributes
2B0h	PROG11_MPSAR	Programmable range 11, start address
2B4h	PROG11_MPEAR	Programmable range 11, end address
2B8h	PROG11_MPPAR	Programmable range 11, memory page protection attributes
2C0h	PROG12_MPSAR	Programmable range 12, start address
2C4h	PROG12_MPEAR	Programmable range 12, end address
2C8h	PROG12_MPPAR	Programmable range 12, memory page protection attributes
2D0h	PROG13_MPSAR	Programmable range 13, start address
2D4h	PROG13_MPEAR	Programmable range 13, end address
2Dh	PROG13_MPPAR	Programmable range 13, memory page protection attributes
2E0h	PROG14_MPSAR	Programmable range 14, start address
2E4h	PROG14_MPEAR	Programmable range 14, end address
2E8h	PROG14_MPPAR	Programmable range 14, memory page protection attributes
2F0h	PROG15_MPSAR	Programmable range 15, start address
2F4h	PROG15_MPEAR	Programmable range 15, end address
2F8h	PROG15_MPPAR	Programmable range 15, memory page protection attributes
300h	FLTADDRR	Fault address
304h	FLTSTAT	Fault status
308h	FLTCLR	Fault clear
End of Table 3-8		

3.2.1.2 Device-Specific MPU Registers

3.2.1.2.1 Configuration Register (CONFIG)

The configuration register (CONFIG) contains the configuration value of the MPU.

Table 3-9 Configuration Register Field Descriptions (Part 1 of 2)

Bits	Field	Description
31 – 24	ADDR_WIDTH	Address alignment for range checking 0 = 1KB alignment 6 = 64KB alignment
23 – 20	NUM_FIXED	Number of fixed address ranges
19 – 16	NUM_PROG	Number of programmable address ranges

Table 3-9 Configuration Register Field Descriptions (Part 2 of 2)

Bits	Field	Description
15 – 12	NUM_AIDS	Number of supported AIDs
11 – 1	Reserved	Reserved. Always read as 0.
0	ASSUME_ALLOWED	Assume allowed bit. When an address is not covered by any MPU protection range, this bit determines whether the transfer is assumed to be allowed or not. 0 = Assume disallowed 1 = Assume allowed
End of Table 3-9		

3.2.2 MPU Programmable Range Registers

3.2.2.1 Programmable Range *n* Start Address Register (PROG*n*_MPSAR)

The Programmable Address Start Register holds the start address for the range. This register is writeable by a supervisor entity only. If NS = 0 (non-secure mode) in the associated MPPAR register, then the register is also writeable only by a secure entity.

The start address must be aligned on a page boundary. The size of the page is 1K byte. The size of the page determines the width of the address field in MPSAR and MPEAR.

Figure 3-1 Programmable Range *n* Start Address Register (PROG*n*_MPSAR)

31	10	9	0
START_ADDR		Reserved	
R/W		R	

Legend: R = Read only; R/W = Read/Write

Table 3-10 Programmable Range *n* Start Address Register Field Descriptions

Bit	Field	Description
31 – 10	START_ADDR	Start address for range n
9 – 0	Reserved	Reserved. Always read as 0.
End of Table 3-10		

Table 3-11 MPU0-MPU5 Programmable Range *n* Start Address Register (PROG*n*_MPSAR) Reset Values (Part 1 of 2)

Register	MPU0	MPU1	MPU2	MPU3	MPU4	MPU5
PROG0_MPSAR	0x01D0_0000	0x23A0_0000	0x02A0_0000	Reserved	Reserved	0x02A0_4000
PROG1_MPSAR	0x01F0_0000	0x23A0_2000	0x02A0_2000	Reserved	Reserved	0x02A0_5000
PROG2_MPSAR	0x02F0_0000	0x023A_6000	0x02A0_6000	Reserved	Reserved	0x02A0_6400
PROG3_MPSAR	0x0200_0000	0x23A0_6800	0x02A0_6800	Reserved	Reserved	0x02A0_7400
PROG4_MPSAR	0x020C_0000	0x23A0_7000	0x02A0_7000	Reserved	Reserved	0x02A0_A000
PROG5_MPSAR	0x021C_0000	0x23A0_8000	0x02A0_8000	Reserved	Reserved	0x02A0_D000
PROG6_MPSAR	0x021D_0000	0x23A0_C000	0x02A0_C000	Reserved	Reserved	0x02A0_E000
PROG7_MPSAR	0x021F_0000	0x23A0_E000	0x02A0_E000	Reserved	Reserved	0x02A0_F000
PROG8_MPSAR	0x0234_0000	0x23A0_F000	0x02A0_F000	Reserved	Reserved	0x02A0_F800
PROG9_MPSAR	0x0254_0000	0x23A0_F800	0x02A0_F800	Reserved	Reserved	0x02A1_2000
PROG10_MPSAR	0x0258_0000	0x23A1_0000	0x02A1_0000	Reserved	Reserved	0x02A1_C000
PROG11_MPSAR	0x0000_0000	0x23A1_C000	0x02A2_0000	Reserved	Reserved	0x02A2_8000
PROG12_MPSAR	0x0290_0000	0x23A4_0000	0x02A4_0000	Reserved	Reserved	0x02A6_0000
PROG13_MPSAR	0x01E8_0000	0x23A8_0000	0x02A8_0000	Reserved	Reserved	0x02AA_0000

PRODUCT PREVIEW

Table 3-11 MPU0-MPU5 Programmable Range *n* Start Address Register (PROG*n*_MPSAR) Reset Values (Part 2 of 2)

Register	MPU0	MPU1	MPU2	MPU3	MPU4	MPU5
PROG14_MPSAR	0x01E8_0800	0x23B0_0000	0x02AC_0000	Reserved	Reserved	0x02B0_0000
PROG15_MPSAR	0x01E0_0000	0x23B8_0000	0x02AE_0000	Reserved	Reserved	0x02B8_0000
End of Table 3-11						

Table 3-12 MPU12-MPU15 Programmable Range *n* Start Address Register (PROG*n*_MPSAR) Reset Values

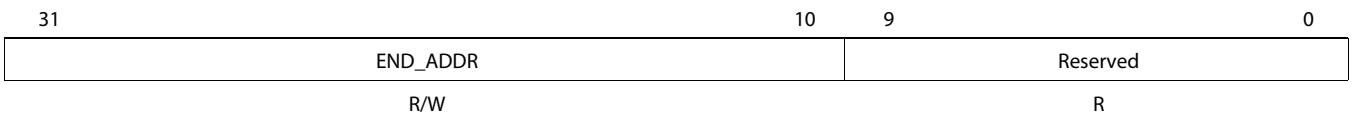
Register	MPU12	MPU13	MPU14	MPU15
PROG0_MPSAR	0x2100_0400	0x2100_0400	0x2100_0800	0x2079_0000
PROG1_MPSAR	0x0000_0000	0x0000_0000	0x0000_0000	0x2060_0000
PROG2_MPSAR	N/A	N/A	N/A	0x206c_0000
PROG3_MPSAR	N/A	N/A	N/A	0x2070_0000
PROG4_MPSAR	N/A	N/A	N/A	0x2078_0000
PROG5_MPSAR	N/A	N/A	N/A	0x0000_0000
PROG6_MPSAR	N/A	N/A	N/A	0x0000_0000
PROG7_MPSAR	N/A	N/A	N/A	0x0000_0000
PROG8_MPSAR	N/A	N/A	N/A	N/A
PROG9_MPSAR	N/A	N/A	N/A	N/A
PROG10_MPSAR	N/A	N/A	N/A	N/A
PROG11_MPSAR	N/A	N/A	N/A	N/A
PROG12_MPSAR	N/A	N/A	N/A	N/A
PROG13_MPSAR	N/A	N/A	N/A	N/A
PROG14_MPSAR	N/A	N/A	N/A	N/A
PROG15_MPSAR	N/A	N/A	N/A	N/A
End of Table 3-12				

3.2.2.2 Programmable Range *n* - End Address Register (PROG*n*_MPEAR)

The programmable address end register holds the end address for the range. This register is writeable by a supervisor entity only. If NS = 0 (non-secure mode) in the associated MPPAR register then the register is also writeable only by a secure entity.

The end address must be aligned on a page boundary. The size of the page depends on the MPU number. The page size for MPU1 is 1K byte and for MPU2 it is 64K bytes. The size of the page determines the width of the address field in MPSAR and MPEAR

Figure 3-2 Programmable Range *n* End Address Register (PROG*n*_MPEAR)



Legend: R = Read only; R/W = Read/Write

Table 3-13 Programmable Range *n* End Address Register Field Descriptions

Bit	Field	Description
31 – 10	END_ADDR	End address for range n
9 – 0	Reserved	Reserved. Always read as 3FFh.
End of Table 3-13		

Table 3-14 MPU0-MPU5 Programmable Range *n* End Address Register (PROG*n*_MPEAR) Reset Values

Register	MPU0	MPU1	MPU2	MPU3	MPU4	MPU5
PROG0_MPEAR	0x01DF_FFFF	0x23A0_1FFF	0x02A0_00FF	Reserved	Reserved	0x02A0_4FFF
PROG1_MPEAR	0x01F7_FFFF	0x23A0_5FFF	0x02A0_3FFF	Reserved	Reserved	0x02A0_5FFF
PROG2_MPEAR	0x02FF_FFFF	0x23A0_67FF	0x02A0_63FF	Reserved	Reserved	0x02A0_67FF
PROG3_MPEAR	0x020B_FFFF	0x23A0_6FFF	0x02A0_6FFF	Reserved	Reserved	0x02A0_7FFF
PROG4_MPEAR	0x020F_FFFF	0x23A0_7FFF	0x02A0_73FF	Reserved	Reserved	0x02A0_BFFF
PROG5_MPEAR	0x021C_83FF	0x23A0_BFFF	0x02A0_9FFF	Reserved	Reserved	0x02A0_DFFF
PROG6_MPEAR	0x021D_C0FF	0x23A0_DFFF	0x02A0_CFFF	Reserved	Reserved	0x02A0_E7FF
PROG7_MPEAR	0x021F_C7FF	0x23A0_EFFF	0x02A0_E7FF	Reserved	Reserved	0x02A0_F7FF
PROG8_MPEAR	0x0234_C0FF	0x23A0_F7FF	0x02A0_F7FF	Reserved	Reserved	0x02A0_FFFF
PROG9_MPEAR	0x0255_FFFF	0x23A0_FFFF	0x02A0_FFFF	Reserved	Reserved	0x02A1_7FFF
PROG10_MPEAR	0x025F_FFFF	0x23A1_BFFF	0x02A1_1FFF	Reserved	Reserved	0x02A1_FFFF
PROG11_MPEAR	0x0000_0000	0x23A3_FFFF	0x02A2_5FFF	Reserved	Reserved	0x02A3_FFFF
PROG12_MPEAR	0x029F_FFFF	0x23A7_FFFF	0x02A5_FFFF	Reserved	Reserved	0x02A7_FFFF
PROG13_MPEAR	0x01E8_07FF	0x23AF_FFFF	0x02A9_FFFF	Reserved	Reserved	0x02AB_FFFF
PROG14_MPEAR	0x01E8_43FF	0x23B7_FFFF	0x02AD_FFFF	Reserved	Reserved	0x02B7_FFFF
PROG15_MPEAR	0x01E7_FFFF	0x23BF_FFFF	0x02AF_FFFF	Reserved	Reserved	0x02BF_FFFF
End of Table 3-14						

Table 3-15 MPU6-MPU11 Programmable Range *n* End Address Register (PROG*n*_MPEAR) Reset Values

Register	MPU6	MPU7	MPU8	MPU9	MPU10	MPU11
PROG0_MPEAR	Reserved	0x2103_FFFF	0x31FF_FFFF	0x0260_1FFF	0x0264_07FF	0x022F_027F
PROG1_MPEAR	Reserved	0x07FF_FFFF	0x33FF_FFFF	0x0260_5FFF	0x0000_0000	0x0231_01FF
PROG2_MPEAR	Reserved	0x0FFF_FFFF	0x35FF_FFFF	0x0260_9FFF	N/A	0x0232_FFFF
PROG3_MPEAR	Reserved	0x17FF_FFFF	0x37FF_FFFF	0x0257_FFFF	N/A	0x0233_07FF
PROG4_MPEAR	Reserved	0x1FFF_FFFF	0x39FF_FFFF	0x0000_0000	N/A	0x0235_0FFF
PROG5_MPEAR	Reserved	0x27FF_FFFF	0x3BFF_FFFF	0x0000_0000	N/A	0x0263_FFFF
PROG6_MPEAR	Reserved	0x2FFF_FFFF	0x3FFF_FFFF	0x0000_0000	N/A	0x024B_3FFF
PROG7_MPEAR	Reserved	0x37FF_FFFF	0x2100_0AFF	0x0000_0000	N/A	0x024C_0BFF
PROG8_MPEAR	Reserved	0x3FFF_FFFF	N/A	0x0000_0000	N/A	0x0250_7FFF
PROG9_MPEAR	Reserved	0x47FF_FFFF	N/A	0x0000_0000	N/A	0x0253_0BFF
PROG10_MPEAR	Reserved	0x4FFF_FFFF	N/A	0x0000_0000	N/A	0x0253_FFFF
PROG11_MPEAR	Reserved	0x57FF_FFFF	N/A	0x0000_0000	N/A	0x0260_BFFF
PROG12_MPEAR	Reserved	0x5FFF_FFFF	N/A	0x0000_0000	N/A	0x0262_0FFF
PROG13_MPEAR	Reserved	0x67FF_FFFF	N/A	0x0000_0000	N/A	0x03FF_FFFF
PROG14_MPEAR	Reserved	0x6FFF_FFFF	N/A	0x0000_0000	N/A	0x021E_1FFF
PROG15_MPEAR	Reserved	0x7FFF_FFFF	N/A	0x0000_0000	N/A	0x026F_FFFF
End of Table 3-15						

PRODUCT PREVIEW

Table 3-16 MPU12-MPU15 Programmable Range *n* End Address Register (PROG*n*_MPEAR) Reset Values

Register	MPU12	MPU13	MPU14	MPU15
PROG0_MPEAR	0x2100_07FF	0x2100_07FF	0x2100_0AFF	0x209F_FFFF
PROG1_MPEAR	0x0000_0000	0x0000_0000	0x0000_0000	0x206B_FFFF
PROG2_MPEAR	N/A	N/A	N/A	0x206F_FFFF
PROG3_MPEAR	N/A	N/A	N/A	0x2077_FFFF
PROG4_MPEAR	N/A	N/A	N/A	0x2078_FFFF
PROG5_MPEAR	N/A	N/A	N/A	0x0000_0000
PROG6_MPEAR	N/A	N/A	N/A	0x0000_0000
PROG7_MPEAR	N/A	N/A	N/A	0x0000_0000
PROG8_MPEAR	N/A	N/A	N/A	N/A
PROG9_MPEAR	N/A	N/A	N/A	N/A
PROG10_MPEAR	N/A	N/A	N/A	N/A
PROG11_MPEAR	N/A	N/A	N/A	N/A
PROG12_MPEAR	N/A	N/A	N/A	N/A
PROG13_MPEAR	N/A	N/A	N/A	N/A
PROG14_MPEAR	N/A	N/A	N/A	N/A
PROG15_MPEAR	N/A	N/A	N/A	N/A
End of Table 3-16				

3.2.2.3 Programmable Range *n* Memory Protection Page Attribute Register (PROG*n*_MPPAR)

The programmable address memory protection page attribute register holds the permissions for the region. This register is writeable only by a non-debug supervisor entity. If NS = 0 (secure mode) then the register is also writeable only by a non-debug secure entity. The NS bit is writeable only by a non-debug secure entity. For debug accesses, the register is writeable only when NS = 1 or EMU = 1.

Figure 3-3 Programmable Range *n* Memory Protection Page Attribute Register (PROG*n*_MPPAR)

31	26	25	24	23	22	21	20	19	18	17	16	15
Reserved		AID15	AID14	AID13	AID12	AID11	AID10	AID9	AID8	AID7	AID6	AID5
R		R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AID4	AID3	AID2	AID1	AID0	AIDX	Reserved	NS	EMU	SR	SW	SX	UR	UW	UX
R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Legend: R = Read only; R/W = Read/Write

Table 3-17 Programmable Range *n* Memory Protection Page Attribute Register Field Descriptions (Part 1 of 3)

Bits	Name	Description
31 – 26	Reserved	Reserved. Always read as 0.
25	AID15	Controls access from ID = 15 0 = Access denied 1 = Access granted
24	AID14	Controls access from ID = 14 0 = Access denied 1 = Access granted
23	AID13	Controls access from ID = 13 0 = Access denied 1 = Access granted

Table 3-17 Programmable Range *n* Memory Protection Page Attribute Register Field Descriptions (Part 2 of 3)

Bits	Name	Description
22	AID12	Controls access from ID = 12 0 = Access denied 1 = Access granted
21	AID11	Controls access from ID = 11 0 = Access denied 1 = Access granted
20	AID10	Controls access from ID = 10 0 = Access denied 1 = Access granted
19	AID9	Controls access from ID = 9 0 = Access denied 1 = Access granted
18	AID8	Controls access from ID = 8 0 = Access denied 1 = Access granted
17	AID7	Controls access from ID = 7 0 = Access denied 1 = Access granted
16	AID6	Controls access from ID = 6 0 = Access denied 1 = Access granted
15	AID5	Controls access from ID = 5 0 = Access denied 1 = Access granted
14	AID4	Controls access from ID = 4 0 = Access denied 1 = Access granted
13	AID3	Controls access from ID = 3 0 = Access denied 1 = Access granted
12	AID2	Controls access from ID = 2 0 = Access denied 1 = Access granted
11	AID1	Controls access from ID = 1 0 = Access denied 1 = Access granted
10	AID0	Controls access from ID = 0 0 = Access denied 1 = Access granted
9	AIDX	Controls access from ID > 15 0 = Access denied 1 = Access granted
8	Reserved	Reserved. Always reads as 0.
7	NS	Non-secure access permission 0 = Only secure access allowed 1 = Non-secure access allowed
6	EMU	Emulation (debug) access permission. This bit is ignored if NS = 1 0 = Debug access not allowed 1 = Debug access allowed
5	SR	Supervisor Read permission 0 = Access not allowed 1 = Access allowed

PRODUCT PREVIEW

Table 3-17 Programmable Range *n* Memory Protection Page Attribute Register Field Descriptions (Part 3 of 3)

Bits	Name	Description
4	SW	Supervisor Write permission 0 = Access not allowed 1 = Access allowed
3	SX	Supervisor Execute permission 0 = Access not allowed 1 = Access allowed
2	UR	User Read permission 0 = Access not allowed 1 = Access allowed
1	UW	User Write permission 0 = Access not allowed 1 = Access allowed
0	UX	User Execute permission 0 = Access not allowed 1 = Access allowed
End of Table 3-171		

Table 3-18 MPU0-MPU5 Programmable Range *n* Memory Protection Page Attribute Register (PROG*n*_MPPAR) Reset Values

Register	MPU0	MPU1	MPU2	MPU3	MPU4	MPU5
PROG0_MPPAR	0x03FF_FCB6	0x03FF_FCB6	0x03FF_FCB6	Reserved	Reserved	0x03FF_FCB4
PROG1_MPPAR	0x03FF_FCB6	0x03FF_FCB4	0x03FF_FCB4	Reserved	Reserved	0x03FF_FCB4
PROG2_MPPAR	0x03FF_FCB6	0x03FF_FCA4	0x03FF_FCA4	Reserved	Reserved	0x03FF_FCA4
PROG3_MPPAR	0x03FF_FCB6	0x03FF_FCB4	0x03FF_FCB4	Reserved	Reserved	0x03FF_FCF4
PROG4_MPPAR	0x03FF_FCB6	0x03FF_FCF4	0x03FF_FCF4	Reserved	Reserved	0x03FF_FCB4
PROG5_MPPAR	0x03FF_FCB6	0x03FF_FCB4	0x03FF_FCB4	Reserved	Reserved	0x03FF_FCB4
PROG6_MPPAR	0x03FF_FCB6	0x03FF_FCB4	0x03FF_FCB4	Reserved	Reserved	0x03FF_FCB4
PROG7_MPPAR	0x03FF_FCB6	0x03FF_FCB4	0x03FF_FCB4	Reserved	Reserved	0x03FF_FCB4
PROG8_MPPAR	0x03FF_FCB6	0x03FF_FCB4	0x03FF_FCB4	Reserved	Reserved	0x03FF_FCF4
PROG9_MPPAR	0x03FF_FCB6	0x03FF_FCF4	0x03FF_FCF4	Reserved	Reserved	0x03FF_FCB4
PROG10_MPPAR	0x03FF_FCB6	0x03FF_FCB4	0x03FF_FCB4	Reserved	Reserved	0x03FF_FCF4
PROG11_MPPAR	0x03FF_FCB6	0x03FF_FCF4	0x03FF_FCF4	Reserved	Reserved	0x03FF_FCF4
PROG12_MPPAR	0x03FF_FCB4	0x03FF_FCA4	0x03FF_FCA4	Reserved	Reserved	0x03FF_FCA4
PROG13_MPPAR	0x03FF_FCB6	0x03FF_FCB6	0x03FF_FCB6	Reserved	Reserved	0x03FF_FCB6
PROG14_MPPAR	0x03FF_FCB0	0x03FF_FCA4	0x03FF_FCB6	Reserved	Reserved	0x03FF_FCA4
PROG15_MPPAR	0x03FF_FCB6	0x03FF_FCA4	0x03FF_FCB6	Reserved	Reserved	0x03FF_FCA4
End of Table 3-18						

PRODUCT PREVIEW

Table 3-19 MPU6-MPU11 Programmable Range *n* Memory Protection Page Attribute Register (PROG*n*_MPPAR) Reset Values

Register	MPU6	MPU7	MPU8	MPU9	MPU10	MPU11
PROG0_MPPAR	Reserved	0x03FF_FCB6	0x03FF_FCBF	0x03FF_FCB6	0x03FF_FCB6	0x03FF_FCB6
PROG1_MPPAR	Reserved	0x03FF_FCBF	0x03FF_FCBF	0x03FF_FCB6	0x03FF_FCB6	0x03FF_FCB0
PROG2_MPPAR	Reserved	0x03FF_FCBF	0x03FF_FCBF	0x03FF_FCB6	N/A	0x03FF_FCB6
PROG3_MPPAR	Reserved	0x03FF_FCBF	0x03FF_FCBF	0x03FF_FCB6	N/A	0x03FF_FCB0
PROG4_MPPAR	Reserved	0x03FF_FCBF	0x03FF_FCBF	0x03FF_FCB6	N/A	0x03FF_FCB0
PROG5_MPPAR	Reserved	0x03FF_FCBF	0x03FF_FCBF	0x03FF_FCB6	N/A	0x03FF_FCB6
PROG6_MPPAR	Reserved	0x03FF_FCBF	0x03FF_FCBF	0x03FF_FCB6	N/A	0x03FF_FCB6
PROG7_MPPAR	Reserved	0x03FF_FCBF	0x03FF_FCB6	0x03FF_FCB6	N/A	0x03FF_FCB0
PROG8_MPPAR	Reserved	0x03FF_FCBF	N/A	0x03FF_FCB6	N/A	0x03FF_FCB0
PROG9_MPPAR	Reserved	0x03FF_FCBF	N/A	0x03FF_FCB6	N/A	0x03FF_FCB6
PROG10_MPPAR	Reserved	0x03FF_FCBF	N/A	0x03FF_FCB6	N/A	0x03FF_FCB6
PROG11_MPPAR	Reserved	0x03FF_FCBF	N/A	0x03FF_FCB6	N/A	0x03FF_FCB6
PROG12_MPPAR	Reserved	0x03FF_FCBF	N/A	0x03FF_FCB6	N/A	0x03FF_FCB0
PROG13_MPPAR	Reserved	0x03FF_FCBF	N/A	0x03FF_FCB6	N/A	0x03FF_FCB6
PROG14_MPPAR	Reserved	0x03FF_FCBF	N/A	0x03FF_FCB6	N/A	0x03FF_FCB0
PROG15_MPPAR	Reserved	0x03FF_FCBF	N/A	0x03FF_FCB6	N/A	0x03FF_FCB6
End of Table 3-19						

Table 3-20 MPU12-MPU15 Programmable Range *n* Memory Protection Page Attribute Register (PROG*n*_MPPAR) Reset Values

Register	MPU12	MPU13	MPU14	MPU15
PROG0_MPPAR	0x03FF_FCB6	0x03FF_FCB6	0x03FF_FCB6	0x03FF_FC80
PROG1_MPPAR	0x03FF_FCBF	0x03FF_FCBF	0x03FF_FCBF	0x03FF_FCB6
PROG2_MPPAR	N/A	N/A	N/A	0x03FF_FCB6
PROG3_MPPAR	N/A	N/A	N/A	0x03FF_FCB6
PROG4_MPPAR	N/A	N/A	N/A	0x03FF_FCB6
PROG5_MPPAR	N/A	N/A	N/A	0x03FF_FCB6
PROG6_MPPAR	N/A	N/A	N/A	0x03FF_FCB6
PROG7_MPPAR	N/A	N/A	N/A	0x03FF_FCB6
PROG8_MPPAR	N/A	N/A	N/A	N/A
PROG9_MPPAR	N/A	N/A	N/A	N/A
PROG10_MPPAR	N/A	N/A	N/A	N/A
PROG11_MPPAR	N/A	N/A	N/A	N/A
PROG12_MPPAR	N/A	N/A	N/A	N/A
PROG13_MPPAR	N/A	N/A	N/A	N/A
PROG14_MPPAR	N/A	N/A	N/A	N/A
PROG15_MPPAR	N/A	N/A	N/A	N/A
End of Table 3-20				

PRODUCT PREVIEW

3.3 Interrupts for AM5K2E04/02

This section discusses the interrupt sources, controller, and topology. Also provided are tables describing the interrupt events.

3.3.1 Interrupt Sources and Interrupt Controller

The ARM CorePac interrupts on the AM5K2E04/02 device is configured through the ARM CorePac Interrupt Controller. It allows for up to 480 system events to be programmed to any of the ARM core's IRQ/FIQ interrupts. In addition error-class events or infrequently used events are also routed through the system event router to offload the ARM CorePac interrupt controller. This is accomplished through the CorePac Interrupt Controller block CIC2. Further, CIC2 provides 8 events each to EDMA3CC0, EDMA3CC1, EDMA3C2, EDMA3CC3, EDMA3CC4, and Hyperlink.

Modules such as CP_MPU, BOOT_CFG, and CP_Tracer have level interrupts and EOI handshaking interface. The EOI value is 0 for CP_MPU, BOOT_CFG, and CP_Tracer.

Figure 3-4 shows the AM5K2E04/02 interrupt topology.

Figure 3-4 Interrupt Topology for AM5K2E04/02

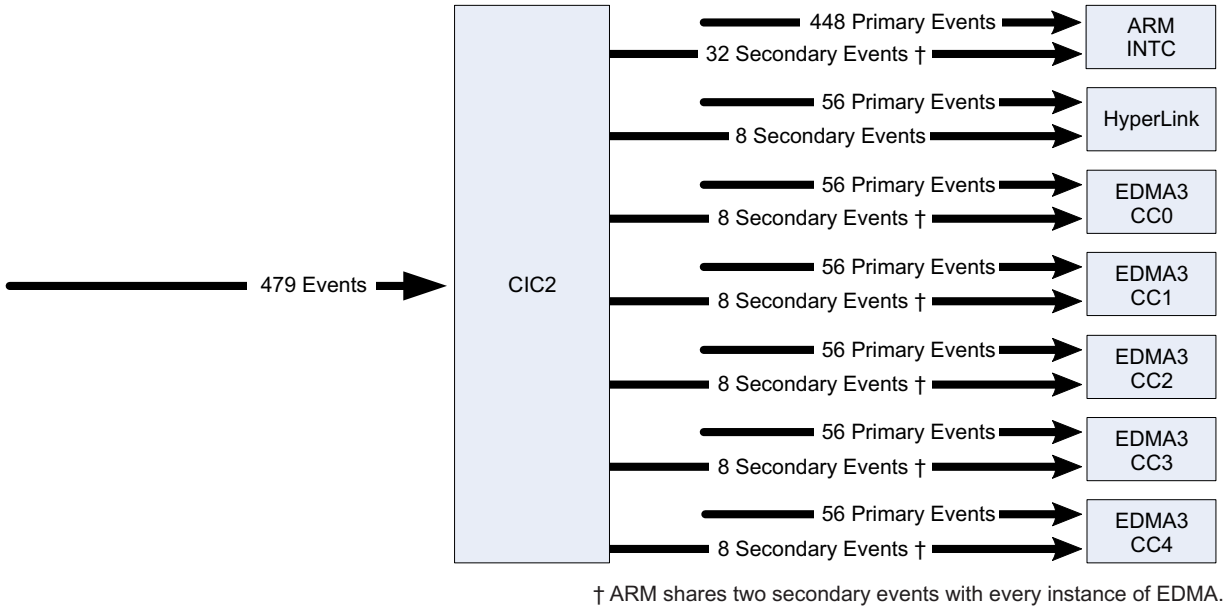


Table 3-21 lists the ARM CorePac event inputs

Table 3-21 System Event Mapping — ARM CorePac Interrupts (Part 1 of 12)

Event No.	Event Name	Description
0	RSTMUX_INT8	Boot config watchdog timer expiration (timer 16) event for ARM core 0
1	RSTMUX_INT9	Boot config watchdog timer expiration (timer 17) event for ARM core 1
2	RSTMUX_INT10	Boot config watchdog timer expiration (timer 18) event for ARM core 2
3	RSTMUX_INT11	Boot config watchdog timer expiration (timer 19) event for ARM core 3
4	IPC_GR8	Boot config IPCG
5	IPC_GR9	Boot config IPCG
6	IPC_GR10	Boot config IPCG
7	IPC_GR11	Boot config IPCG

Table 3-21 System Event Mapping — ARM CorePac Interrupts (Part 2 of 12)

Event No.	Event Name	Description
8	SEM_INT8	Semaphore interrupt
9	SEM_INT9	Semaphore interrupt
10	SEM_INT10	Semaphore interrupt
11	SEM_INT11	Semaphore interrupt
12	SEM_ERR8	Semaphore error interrupt
13	SEM_ERR9	Semaphore error interrupt
14	SEM_ERR10	Semaphore error interrupt
15	SEM_ERR11	Semaphore error interrupt
16	MSMC_MPF_ERROR8	Memory protection fault indicators for system master PrivID = 8
17	MSMC_MPF_ERROR9	Memory protection fault indicators for system master PrivID = 9
18	MSMC_MPF_ERROR10	Memory protection fault indicators for system master PrivID = 10
19	MSMC_MPF_ERROR11	Memory protection fault indicators for system master PrivID = 11
20	ARM_NPMUIRQ0	ARM performance monitoring unit interrupt request
21	ARM_NPMUIRQ1	ARM performance monitoring unit interrupt request
22	ARM_NPMUIRQ2	ARM performance monitoring unit interrupt request
23	ARM_NPMUIRQ3	ARM performance monitoring unit interrupt request
24	ARM_NINTERRIRQ	ARM internal memory ECC error interrupt request
25	ARM_NAXIERRIRQ	ARM bus error interrupt request
26	PCIE_0_INT0	PCIE0 legacy INTA interrupt
27	PCIE_0_INT1	PCIE0 legacy INTB interrupt
28	PCIE_0_INT2	PCIE0 legacy INTC interrupt
29	PCIE_0_INT3	PCIE0 legacy INTD interrupt
30	PCIE_0_INT4	PCIE0 MSI interrupt
31	PCIE_0_INT5	PCIE0 MSI interrupt
32	PCIE_0_INT6	PCIE0 MSI interrupt
33	PCIE_0_INT7	PCIE0 MSI interrupt
34	PCIE_0_INT8	PCIE0 MSI interrupt
35	PCIE_0_INT9	PCIE0 MSI interrupt
36	PCIE_0_INT10	PCIE0 MSI interrupt
37	PCIE_0_INT11	PCIE0 MSI interrupt
38	PCIE_0_INT12	PCIE0 error interrupt
39	PCIE_0_INT13	PCIE0 power management interrupt
40	QMSS_QUE_PEND_658	Navigator transmit queue pending event for indicated queue
41	QMSS_QUE_PEND_659	Navigator transmit queue pending event for indicated queue
42	QMSS_QUE_PEND_660	Navigator transmit queue pending event for indicated queue
43	QMSS_QUE_PEND_661	Navigator transmit queue pending event for indicated queue
44	QMSS_QUE_PEND_662	Navigator transmit queue pending event for indicated queue
45	QMSS_QUE_PEND_663	Navigator transmit queue pending event for indicated queue
46	QMSS_QUE_PEND_664	Navigator transmit queue pending event for indicated queue
47	QMSS_QUE_PEND_665	Navigator transmit queue pending event for indicated queue
48	QMSS_QUE_PEND_528	Navigator transmit queue pending event for indicated queue
49	QMSS_QUE_PEND_529	Navigator transmit queue pending event for indicated queue
50	QMSS_QUE_PEND_530	Navigator transmit queue pending event for indicated queue
51	QMSS_QUE_PEND_531	Navigator transmit queue pending event for indicated queue

PRODUCT PREVIEW

Table 3-21 System Event Mapping — ARM CorePac Interrupts (Part 3 of 12)

Event No.	Event Name	Description
52	QMSS_QUE_PEND_532	Navigator transmit queue pending event for indicated queue
53	QMSS_QUE_PEND_533	Navigator transmit queue pending event for indicated queue
54	QMSS_QUE_PEND_534	Navigator transmit queue pending event for indicated queue
55	QMSS_QUE_PEND_535	Navigator transmit queue pending event for indicated queue
56	QMSS_QUE_PEND_536	Navigator transmit queue pending event for indicated queue
57	QMSS_QUE_PEND_537	Navigator transmit queue pending event for indicated queue
58	QMSS_QUE_PEND_538	Navigator transmit queue pending event for indicated queue
59	QMSS_QUE_PEND_539	Navigator transmit queue pending event for indicated queue
60	QMSS_QUE_PEND_540	Navigator transmit queue pending event for indicated queue
61	QMSS_QUE_PEND_541	Navigator transmit queue pending event for indicated queue
62	QMSS_QUE_PEND_542	Navigator transmit queue pending event for indicated queue
63	QMSS_QUE_PEND_543	Navigator transmit queue pending event for indicated queue
64	QMSS_QUE_PEND_544	Navigator transmit queue pending event for indicated queue
65	QMSS_QUE_PEND_545	Navigator transmit queue pending event for indicated queue
66	QMSS_QUE_PEND_546	Navigator transmit queue pending event for indicated queue
67	QMSS_QUE_PEND_547	Navigator transmit queue pending event for indicated queue
68	QMSS_QUE_PEND_548	Navigator transmit queue pending event for indicated queue
69	QMSS_QUE_PEND_549	Navigator transmit queue pending event for indicated queue
70	QMSS_QUE_PEND_550	Navigator transmit queue pending event for indicated queue
71	QMSS_QUE_PEND_551	Navigator transmit queue pending event for indicated queue
72	QMSS_QUE_PEND_552	Navigator transmit queue pending event for indicated queue
73	QMSS_QUE_PEND_553	Navigator transmit queue pending event for indicated queue
74	QMSS_QUE_PEND_554	Navigator transmit queue pending event for indicated queue
75	QMSS_QUE_PEND_555	Navigator transmit queue pending event for indicated queue
76	QMSS_QUE_PEND_556	Navigator transmit queue pending event for indicated queue
77	QMSS_QUE_PEND_557	Navigator transmit queue pending event for indicated queue
78	QMSS_QUE_PEND_558	Navigator transmit queue pending event for indicated queue
79	QMSS_QUE_PEND_559	Navigator transmit queue pending event for indicated queue
80	Reserved	Reserved
81	Reserved	Reserved
82	USIM_PONIRQ	USIM interrupt
83	USIM_RREQ	USIM RX DMA event
84	USIM_WREQ	USIM TX DMA event
85	TSIP_RCV_FINT0	TSIP receive frame interrupt for channel 0
86	TSIP_XMT_FINT0	TSIP transmit frame interrupt for channel 0
87	TSIP_RCV_SFINT0	TSIP receive super frame interrupt for channel 0
88	TSIP_XMT_SFINT0	TSIP transmit super frame interrupt for channel 0
89	TSIP_EINT0	TSIP error interrupt for channel 0
90	TSIP_RCV_FINT1	TSIP receive frame interrupt for channel 1
91	TSIP_XMT_FINT1	TSIP transmit frame interrupt for channel 1
92	TSIP_RCV_SFINT1	TSIP receive super frame interrupt for channel 1
93	TSIP_XMT_SFINT1	TSIP transmit super frame interrupt for channel 1
94	TSIP_EINT1	TSIP error interrupt for channel 1
95	Reserved	Reserved

Table 3-21 System Event Mapping — ARM CorePac Interrupts (Part 4 of 12)

Event No.	Event Name	Description
96	TIMER_8_INTL	Timer interrupt low
97	TIMER_8_INTH	Timer interrupt high
98	TIMER_9_INTL	Timer interrupt low
99	TIMER_9_INTH	Timer interrupt high
100	TIMER_10_INTL	Timer interrupt low
101	TIMER_10_INTH	Timer interrupt high
102	TIMER_11_INTL	Timer interrupt low
103	TIMER_11_INTH	Timer interrupt high
104	TIMER_12_INTL	Timer interrupt low
105	TIMER_12_INTH	Timer interrupt high
106	TIMER_13_INTL	Timer interrupt low
107	TIMER_13_INTH	Timer interrupt high
108	TIMER_14_INTL	Timer interrupt low
109	TIMER_14_INTH	Timer interrupt high
110	TIMER_15_INTL	Timer interrupt low
111	TIMER_15_INTH	Timer interrupt high
112	TIMER_16_INTL	Timer interrupt low
113	TIMER_16_INTH	Timer interrupt high
114	TIMER_17_INTL	Timer interrupt low
115	TIMER_17_INTH	Timer interrupt high
116	TIMER_18_INTL	Timer interrupt low
117	TIMER_18_INTH	Timer interrupt high
118	TIMER_19_INTL	Timer interrupt low
119	TIMER_19_INTH	Timer interrupt high
120	GPIO_INT0	GPIO interrupt
121	GPIO_INT1	GPIO interrupt
122	GPIO_INT2	GPIO interrupt
123	GPIO_INT3	GPIO interrupt
124	GPIO_INT4	GPIO interrupt
125	GPIO_INT5	GPIO interrupt
126	GPIO_INT6	GPIO interrupt
127	GPIO_INT7	GPIO interrupt
128	GPIO_INT8	GPIO interrupt
129	GPIO_INT9	GPIO interrupt
130	GPIO_INT10	GPIO interrupt
131	GPIO_INT11	GPIO interrupt
132	GPIO_INT12	GPIO interrupt
133	GPIO_INT13	GPIO interrupt
134	GPIO_INT14	GPIO interrupt
135	GPIO_INT15	GPIO interrupt
136	GPIO_INT16	GPIO interrupt
137	GPIO_INT17	GPIO interrupt
138	GPIO_INT18	GPIO interrupt
139	GPIO_INT19	GPIO interrupt

Table 3-21 System Event Mapping — ARM CorePac Interrupts (Part 5 of 12)

Event No.	Event Name	Description
140	GPIO_INT20	GPIO interrupt
141	GPIO_INT21	GPIO interrupt
142	GPIO_INT22	GPIO interrupt
143	GPIO_INT23	GPIO interrupt
144	GPIO_INT24	GPIO interrupt
145	GPIO_INT25	GPIO interrupt
146	GPIO_INT26	GPIO interrupt
147	GPIO_INT27	GPIO interrupt
148	GPIO_INT28	GPIO interrupt
149	GPIO_INT29	GPIO interrupt
150	GPIO_INT30	GPIO interrupt
151	GPIO_INT31	GPIO interrupt
152	USB_0_INT00	USB 0 event ring 0 interrupt
153	USB_0_INT01	USB 0 event ring 1 interrupt
154	USB_0_INT02	USB 0 event ring 2 interrupt
155	USB_0_INT03	USB 0 event ring 3 interrupt
156	USB_0_INT04	USB 0 event ring 4 interrupt
157	USB_0_INT05	USB 0 event ring 5 interrupt
158	USB_0_INT06	USB 0 event ring 6 interrupt
159	USB_0_INT07	USB 0 event ring 7 interrupt
160	USB_0_INT08	USB 0 event ring 8 interrupt
161	USB_0_INT09	USB 0 event ring 9 interrupt
162	USB_0_INT10	USB 0 event ring 10 interrupt
163	USB_0_INT11	USB 0 event ring 11 interrupt
164	USB_0_INT12	USB 0 event ring 12 interrupt
165	USB_0_INT13	USB 0 event ring 13 interrupt
166	USB_0_INT14	USB 0 event ring 14 interrupt
167	USB_0_INT15	USB 0 event ring 15 interrupt
168	USB_0_OABSINT	USB 0 OABS interrupt
169	USB_0_MISCINT	USB0_misc_int
170	Reserved	Reserved
171	Reserved	Reserved
172	Reserved	Reserved
173	Reserved	Reserved
174	Reserved	Reserved
175	Reserved	Reserved
176	QMSS1_ECC_INTR	Navigator ECC error interrupt
177	QMSS_INTD_1_PKTDMA_0	Navigator interrupt for Packet DMA starvation
178	QMSS_INTD_1_PKTDMA_1	Navigator interrupt for Packet DMA starvation
179	QMSS_INTD_1_HIGH_0	Navigator hi interrupt
180	QMSS_INTD_1_HIGH_1	Navigator hi interrupt
181	QMSS_INTD_1_HIGH_2	Navigator hi interrupt
182	QMSS_INTD_1_HIGH_3	Navigator hi interrupt
183	QMSS_INTD_1_HIGH_4	Navigator hi interrupt

Table 3-21 System Event Mapping — ARM CorePac Interrupts (Part 6 of 12)

Event No.	Event Name	Description
184	QMSS_INTD_1_HIGH_5	Navigator hi interrupt
185	QMSS_INTD_1_HIGH_6	Navigator hi interrupt
186	QMSS_INTD_1_HIGH_7	Navigator hi interrupt
187	QMSS_INTD_1_HIGH_8	Navigator hi interrupt
188	QMSS_INTD_1_HIGH_9	Navigator hi interrupt
189	QMSS_INTD_1_HIGH_10	Navigator hi interrupt
190	QMSS_INTD_1_HIGH_11	Navigator hi interrupt
191	QMSS_INTD_1_HIGH_12	Navigator hi interrupt
192	QMSS_INTD_1_HIGH_13	Navigator hi interrupt
193	QMSS_INTD_1_HIGH_14	Navigator hi interrupt
194	QMSS_INTD_1_HIGH_15	Navigator hi interrupt
195	QMSS_INTD_1_HIGH_16	Navigator hi interrupt
196	QMSS_INTD_1_HIGH_17	Navigator hi interrupt
197	QMSS_INTD_1_HIGH_18	Navigator hi interrupt
198	QMSS_INTD_1_HIGH_19	Navigator hi interrupt
199	QMSS_INTD_1_HIGH_20	Navigator hi interrupt
200	QMSS_INTD_1_HIGH_21	Navigator hi interrupt
201	QMSS_INTD_1_HIGH_22	Navigator hi interrupt
202	QMSS_INTD_1_HIGH_23	Navigator hi interrupt
203	QMSS_INTD_1_HIGH_24	Navigator hi interrupt
204	QMSS_INTD_1_HIGH_25	Navigator hi interrupt
205	QMSS_INTD_1_HIGH_26	Navigator hi interrupt
206	QMSS_INTD_1_HIGH_27	Navigator hi interrupt
207	QMSS_INTD_1_HIGH_28	Navigator hi interrupt
208	QMSS_INTD_1_HIGH_29	Navigator hi interrupt
209	QMSS_INTD_1_HIGH_30	Navigator hi interrupt
210	QMSS_INTD_1_HIGH_31	Navigator hi interrupt
211	QMSS_INTD_1_LOW_0	Navigator interrupt
212	QMSS_INTD_1_LOW_1	Navigator interrupt
213	QMSS_INTD_1_LOW_2	Navigator interrupt
214	QMSS_INTD_1_LOW_3	Navigator interrupt
215	QMSS_INTD_1_LOW_4	Navigator interrupt
216	QMSS_INTD_1_LOW_5	Navigator interrupt
217	QMSS_INTD_1_LOW_6	Navigator interrupt
218	QMSS_INTD_1_LOW_7	Navigator interrupt
219	QMSS_INTD_1_LOW_8	Navigator interrupt
220	QMSS_INTD_1_LOW_9	Navigator interrupt
221	QMSS_INTD_1_LOW_10	Navigator interrupt
222	QMSS_INTD_1_LOW_11	Navigator interrupt
223	QMSS_INTD_1_LOW_12	Navigator interrupt
224	QMSS_INTD_1_LOW_13	Navigator interrupt
225	QMSS_INTD_1_LOW_14	Navigator interrupt
226	QMSS_INTD_1_LOW_15	Navigator interrupt
227	Reserved	Reserved

PRODUCT PREVIEW

Table 3-21 System Event Mapping — ARM CorePac Interrupts (Part 7 of 12)

Event No.	Event Name	Description
228	Reserved	Reserved
229	QMSS_INTD_2_HIGH_0	Navigator second hi interrupt
230	QMSS_INTD_2_HIGH_1	Navigator second hi interrupt
231	QMSS_INTD_2_HIGH_2	Navigator second hi interrupt
232	QMSS_INTD_2_HIGH_3	Navigator second hi interrupt
233	QMSS_INTD_2_HIGH_4	Navigator second hi interrupt
234	QMSS_INTD_2_HIGH_5	Navigator second hi interrupt
235	QMSS_INTD_2_HIGH_6	Navigator second hi interrupt
236	QMSS_INTD_2_HIGH_7	Navigator second hi interrupt
237	QMSS_INTD_2_HIGH_8	Navigator second hi interrupt
238	QMSS_INTD_2_HIGH_9	Navigator second hi interrupt
239	QMSS_INTD_2_HIGH_10	Navigator second hi interrupt
240	QMSS_INTD_2_HIGH_11	Navigator second hi interrupt
241	QMSS_INTD_2_HIGH_12	Navigator second hi interrupt
242	QMSS_INTD_2_HIGH_13	Navigator second hi interrupt
243	QMSS_INTD_2_HIGH_14	Navigator second hi interrupt
244	QMSS_INTD_2_HIGH_15	Navigator second hi interrupt
245	QMSS_INTD_2_HIGH_16	Navigator second hi interrupt
246	QMSS_INTD_2_HIGH_17	Navigator second hi interrupt
247	QMSS_INTD_2_HIGH_18	Navigator second hi interrupt
248	QMSS_INTD_2_HIGH_19	Navigator second hi interrupt
249	QMSS_INTD_2_HIGH_20	Navigator second hi interrupt
250	QMSS_INTD_2_HIGH_21	Navigator second hi interrupt
251	QMSS_INTD_2_HIGH_22	Navigator second hi interrupt
252	QMSS_INTD_2_HIGH_23	Navigator second hi interrupt
253	QMSS_INTD_2_HIGH_24	Navigator second hi interrupt
254	QMSS_INTD_2_HIGH_25	Navigator second hi interrupt
255	QMSS_INTD_2_HIGH_26	Navigator second hi interrupt
256	QMSS_INTD_2_HIGH_27	Navigator second hi interrupt
257	QMSS_INTD_2_HIGH_28	Navigator second hi interrupt
258	QMSS_INTD_2_HIGH_29	Navigator second hi interrupt
259	QMSS_INTD_2_HIGH_30	Navigator second hi interrupt
260	QMSS_INTD_2_HIGH_31	Navigator second hi interrupt
261	QMSS_INTD_2_LOW_0	Navigator second interrupt
262	QMSS_INTD_2_LOW_1	Navigator second interrupt
263	QMSS_INTD_2_LOW_2	Navigator second interrupt
264	QMSS_INTD_2_LOW_3	Navigator second interrupt
265	QMSS_INTD_2_LOW_4	Navigator second interrupt
266	QMSS_INTD_2_LOW_5	Navigator second interrupt
267	QMSS_INTD_2_LOW_6	Navigator second interrupt
268	QMSS_INTD_2_LOW_7	Navigator second interrupt
269	QMSS_INTD_2_LOW_8	Navigator second interrupt
270	QMSS_INTD_2_LOW_9	Navigator second interrupt
271	QMSS_INTD_2_LOW_10	Navigator second interrupt

Table 3-21 System Event Mapping — ARM CorePac Interrupts (Part 8 of 12)

Event No.	Event Name	Description
272	QMSS_INTD_2_LOW_11	Navigator second interrupt
273	QMSS_INTD_2_LOW_12	Navigator second interrupt
274	QMSS_INTD_2_LOW_13	Navigator second interrupt
275	QMSS_INTD_2_LOW_14	Navigator second interrupt
276	QMSS_INTD_2_LOW_15	Navigator second interrupt
277	UART_0_UARTINT	UART0 interrupt
278	UART_0_URXEVT	UART0 receive event
279	UART_0_UTXEVT	UART0 transmit event
280	UART_1_UARTINT	UART1 interrupt
281	UART_1_URXEVT	UART1 receive event
282	UART_1_UTXEVT	UART1 transmit event
283	I2C_0_INT	I2C interrupt
284	I2C_0_REVT	I2C receive event
285	I2C_0_XEVT	I2C transmit event
286	I2C_1_INT	I2C interrupt
287	I2C_1_REVT	I2C receive event
288	I2C_1_XEVT	I2C transmit event
289	I2C_2_INT	I2C interrupt
290	I2C_2_REVT	I2C receive event
291	I2C_2_XEVT	I2C transmit event
292	SPI_0_INT0	SPI interrupt
293	SPI_0_INT1	SPI interrupt
294	SPI_0_XEVT	SPI DMA TX event
295	SPI_0_REVT	SPI DMA RX event
296	SPI_1_INT0	SPI interrupt
297	SPI_1_INT1	SPI interrupt
298	SPI_1_XEVT	SPI DMA TX event
299	SPI_1_REVT	SPI DMA RX event
300	SPI_2_INT0	SPI interrupt
301	SPI_2_INT1	SPI interrupt
302	SPI_2_XEVT	SPI DMA TX event
303	SPI_2_REVT	SPI DMA RX event
304	DBGTBR_DMAINT	Debug trace buffer (TBR) DMA event
305	DBGTBR_ACQCOMP	Debug Trace buffer (TBR) acquisition has been completed
306	ARM_TBR_DMA	ARM Trace Buffer (TBR) DMA event
307	ARM_TBR_ACQ	ARM Trace Buffer (TBR) Acquisition has been completed
308	NETCP_0_MDIO_LINK_INT0	Packet Accelerator 1subsystem MDIO interrupt
309	NETCP_0_MDIO_LINK_INT1	Packet Accelerator 1subsystem MDIO interrupt
310	NETCP_0_MDIO_USER_INT0	Packet Accelerator 1subsystem MDIO interrupt
311	NETCP_0_MDIO_USER_INT1	Packet Accelerator 1subsystem MDIO interrupt
312	NETCP_0_MISC_INT	Packet Accelerator 1subsystem MDIO interrupt
313	NETCP_0_PKTDMA_INT0	Packet Accelerator 0 Packet DMA starvation interrupt
314	EDMACC_0_GINT	EDMA3CC0 global completion interrupt
315	EDMACC_0_TC_0_INT	EDMA3CC0 individual completion interrupt

Table 3-21 System Event Mapping — ARM CorePac Interrupts (Part 9 of 12)

Event No.	Event Name	Description
316	EDMACC_0_TC_1_INT	EDMA3CC0 individual completion interrupt
317	EDMACC_0_TC_2_INT	EDMA3CC0 individual completion interrupt
318	EDMACC_0_TC_3_INT	EDMA3CC0 individual completion interrupt
319	EDMACC_0_TC_4_INT	EDMA3CC0 individual completion interrupt
320	EDMACC_0_TC_5_INT	EDMA3CC0 individual completion interrupt
321	EDMACC_0_TC_6_INT	EDMA3CC0 individual completion interrupt
322	EDMACC_0_TC_7_INT	EDMA3CC0 individual completion interrupt
323	EDMACC_1_GINT	EDMA3CC1 global completion interrupt
324	EDMACC_1_TC_0_INT	EDMA3CC1 individual completion interrupt
325	EDMACC_1_TC_1_INT	EDMA3CC1 individual completion interrupt
326	EDMACC_1_TC_2_INT	EDMA3CC1 individual completion interrupt
327	EDMACC_1_TC_3_INT	EDMA3CC1 individual completion interrupt
328	EDMACC_1_TC_4_INT	EDMA3CC1 individual completion interrupt
329	EDMACC_1_TC_5_INT	EDMA3CC1 individual completion interrupt
330	EDMACC_1_TC_6_INT	EDMA3CC1 individual completion interrupt
331	EDMACC_1_TC_7_INT	EDMA3CC1 individual completion interrupt
332	EDMACC_2_GINT	EDMA3CC2 global completion interrupt
333	EDMACC_2_TC_0_INT	EDMA3CC2 individual completion interrupt
334	EDMACC_2_TC_1_INT	EDMA3CC2 individual completion interrupt
335	EDMACC_2_TC_2_INT	EDMA3CC2 individual completion interrupt
336	EDMACC_2_TC_3_INT	EDMA3CC2 individual completion interrupt
337	EDMACC_2_TC_4_INT	EDMA3CC2 individual completion interrupt
338	EDMACC_2_TC_5_INT	EDMA3CC2 individual completion interrupt
339	EDMACC_2_TC_6_INT	EDMA3CC2 individual completion interrupt
340	EDMACC_2_TC_7_INT	EDMA3CC2 individual completion interrupt
341	EDMACC_3_GINT	EDMA3CC3 global completion interrupt
342	EDMACC_3_TC_0_INT	EDMA3CC3 individual completion interrupt
343	EDMACC_3_TC_1_INT	EDMA3CC3 individual completion interrupt
344	EDMACC_3_TC_2_INT	EDMA3CC3 individual completion interrupt
345	EDMACC_3_TC_3_INT	EDMA3CC3 individual completion interrupt
346	EDMACC_3_TC_4_INT	EDMA3CC3 individual completion interrupt
347	EDMACC_3_TC_5_INT	EDMA3CC3 individual completion interrupt
348	EDMACC_3_TC_6_INT	EDMA3CC3 individual completion interrupt
349	EDMACC_3_TC_7_INT	EDMA3CC3 individual completion interrupt
350	EDMACC_4_GINT	EDMA3CC4 global completion interrupt
351	EDMACC_4_TC_0_INT	EDMA3CC4 individual completion interrupt
352	EDMACC_4_TC_1_INT	EDMA3CC4 individual completion interrupt
353	EDMACC_4_TC_2_INT	EDMA3CC4 individual completion interrupt
354	EDMACC_4_TC_3_INT	EDMA3CC4 individual completion interrupt
355	EDMACC_4_TC_4_INT	EDMA3CC4 individual completion interrupt
356	EDMACC_4_TC_5_INT	EDMA3CC4 individual completion interrupt
357	EDMACC_4_TC_6_INT	EDMA3CC4 individual completion interrupt
358	EDMACC_4_TC_7_INT	EDMA3CC4 individual completion interrupt
359	SR_0_PO_VCON_SMPSERR_INT	SmartReflex SMPS error interrupt

Table 3-21 System Event Mapping — ARM CorePac Interrupts (Part 10 of 12)

Event No.	Event Name	Description
360	SR_0_SMARTREFLEX_INTREQ0	SmartReflex controller interrupt
361	SR_0_SMARTREFLEX_INTREQ1	SmartReflex controller interrupt
362	SR_0_SMARTREFLEX_INTREQ2	SmartReflex controller interrupt
363	SR_0_SMARTREFLEX_INTREQ3	SmartReflex controller interrupt
364	SR_0_VPNOSMPSACK	SmartReflex VPVOLTUPDATE has been asserted, but SMPS has not been responded to in a defined time interval
365	SR_0_VPEQVALUE	SmartReflex SRSINTERUPT is asserted, but the new voltage is not different from the current SMPS voltage
366	SR_0_VPMAXVDD	SmartReflex. The new voltage required is equal to or greater than MaxVdd
367	SR_0_VPMINVDD	SmartReflex. The new voltage required is equal to or less than MinVdd
368	SR_0_VPINIDLE	SmartReflex indicating that the FSM of voltage processor is in idle
369	SR_0_VPOPPCHANGEDONE	SmartReflex indicating that the average frequency error is within the desired limit
370	SR_0_VPSMPSACK	SmartReflex VPVOLTUPDATE asserted and SMPS has acknowledged in a defined time interval
371	SR_0_SR_TEMPSENSOR	SmartReflex temperature threshold crossing interrupt
372	SR_0_SR_TIMERINT	SmartReflex internal timer expiration interrupt
373	PCIE_1_INT0	PCIE1 legacy INTA interrupt
374	PCIE_1_INT1	PCIE1 legacy INTB interrupt
375	PCIE_1_INT2	PCIE1 legacy INTC interrupt
376	PCIE_1_INT3	PCIE1 legacy INTD interrupt
377	PCIE_1_INT4	PCIE1 MSI interrupt
378	PCIE_1_INT5	PCIE1 MSI interrupt
379	PCIE_1_INT6	PCIE1 MSI interrupt
380	PCIE_1_INT7	PCIE1 MSI interrupt
381	PCIE_1_INT8	PCIE1 MSI interrupt
382	PCIE_1_INT9	PCIE1 MSI interrupt
383	PCIE_1_INT10	PCIE1 MSI interrupt
384	PCIE_1_INT11	PCIE1 MSI interrupt
385	PCIE_1_INT12	PCIE1 error interrupt
386	PCIE_1_INT13	PCIE1 power management interrupt
387	HYPERLINK_0_INT	HyperLink interrupt
388	Reserved	
389	ARM_NCTIIRQ0	ARM cross trigger (CTI) IRQ interrupt
390	ARM_NCTIIRQ1	ARM cross trigger (CTI) IRQ interrupt
391	ARM_NCTIIRQ2	ARM cross trigger (CTI) IRQ interrupt
392	ARM_NCTIIRQ3	ARM cross trigger (CTI) IRQ interrupt
393	Reserved	Reserved
394	Reserved	Reserved
395	Reserved	Reserved
396	Reserved	Reserved
397	Reserved	Reserved
398	Reserved	Reserved
399	Reserved	Reserved
400	Reserved	Reserved
401	Reserved	Reserved
402	10GbE_LINK_INT0	10 Gigabit Ethernet subsystem MDIO interrupt

PRODUCT PREVIEW

Table 3-21 System Event Mapping — ARM CorePac Interrupts (Part 11 of 12)

Event No.	Event Name	Description
403	10GbE_USER_INT0	10 Gigabit Ethernet subsystem MDIO interrupt
404	10GbE_LINK_INT1	10 Gigabit Ethernet subsystem MDIO interrupt
405	10GbE_USER_INT1	10 Gigabit Ethernet subsystem MDIO interrupt
406	10GbE_MISC_INT	10 Gigabit Ethernet subsystem MDIO interrupt
407	10GbE_INT_PKTDMA_0	10 Gigabit Ethernet Packet DMA starvation interrupt
408	NETCP_1_MDIO_LINK_INT0	Packet Accelerator 1subsystem MDIO interrupt
409	NETCP_1_MDIO_LINK_INT1	Packet Accelerator 1subsystem MDIO interrupt
410	NETCP_1_MDIO_USER_INT0	Packet Accelerator 1subsystem MDIO interrupt
411	NETCP_1_MDIO_USER_INT1	Packet Accelerator 1subsystem MDIO interrupt
412	NETCP_1_MISC_INT	Packet Accelerator 1subsystem MDIO interrupt
413	NETCP_1_PKTDMA_INT0	Packet Accelerator1 Packet DMA starvation interrupt
414	USB_1_INT00	USB 1 event ring 0 interrupt
415	USB_1_INT01	USB 1 event ring 1 interrupt
416	USB_1_INT02	USB 1 event ring 2 interrupt
417	USB_1_INT03	USB 1 event ring 3 interrupt
418	USB_1_INT04	USB 1 event ring 4 interrupt
419	USB_1_INT05	USB 1 event ring 5 interrupt
420	USB_1_INT06	USB 1 event ring 6 interrupt
421	USB_1_INT07	USB 1 event ring 7 interrupt
422	USB_1_INT08	USB 1 event ring 8 interrupt
423	USB_1_INT09	USB 1 event ring 9 interrupt
424	USB_1_INT10	USB 1 event ring 10 interrupt
425	USB_1_INT11	USB 1 event ring 11 interrupt
426	USB_1_INT12	USB 1 event ring 12 interrupt
427	USB_1_INT13	USB 1 event ring 13 interrupt
428	USB_1_INT14	USB 1 event ring 14 interrupt
429	USB_1_INT15	USB 1 event ring 15 interrupt
430	USB_1_OABSINT	USB 1 OABS interrupt
431	USB_1_MISCINT	USB 1 miscellaneous interrupt
432	Reserved	Reserved
433	Reserved	Reserved
434	Reserved	Reserved
435	Reserved	Reserved
436	Reserved	Reserved
437	Reserved	Reserved
438	Reserved	Reserved
439	Reserved	Reserved
440	Reserved	Reserved
441	Reserved	Reserved
442	Reserved	Reserved
443	Reserved	Reserved
444	Reserved	Reserved
445	Reserved	Reserved
446	Reserved	Reserved

Table 3-21 System Event Mapping — ARM CorePac Interrupts (Part 12 of 12)

Event No.	Event Name	Description
447	Reserved	Reserved
448	CIC_2_OUT29	CIC2 interrupt
449	CIC_2_OUT30	CIC2 interrupt
450	CIC_2_OUT31	CIC2 interrupt
451	CIC_2_OUT32	CIC2 interrupt
452	CIC_2_OUT33	CIC2 interrupt
453	CIC_2_OUT34	CIC2 interrupt
454	CIC_2_OUT35	CIC2 interrupt
455	CIC_2_OUT36	CIC2 interrupt
456	CIC_2_OUT37	CIC2 interrupt
457	CIC_2_OUT38	CIC2 interrupt
458	CIC_2_OUT39	CIC2 interrupt
459	CIC_2_OUT40	CIC2 interrupt
460	CIC_2_OUT41	CIC2 interrupt
461	CIC_2_OUT42	CIC2 interrupt
462	CIC_2_OUT43	CIC2 interrupt
463	CIC_2_OUT44	CIC2 interrupt
464	CIC_2_OUT45	CIC2 interrupt
465	CIC_2_OUT46	CIC2 interrupt
466	CIC_2_OUT47	CIC2 interrupt
467	CIC_2_OUT18	CIC2 interrupt
468	CIC_2_OUT19	CIC2 interrupt
469	CIC_2_OUT22	CIC2 interrupt
470	CIC_2_OUT23	CIC2 interrupt
471	CIC_2_OUT50	CIC2 interrupt
472	CIC_2_OUT51	CIC2 interrupt
473	CIC_2_OUT66	CIC2 interrupt
474	CIC_2_OUT67	CIC2 interrupt
475	CIC_2_OUT88	CIC2 interrupt
476	CIC_2_OUT89	CIC2 interrupt
477	CIC_2_OUT90	CIC2 interrupt
478	CIC_2_OUT91	CIC2 interrupt
479	CIC_2_OUT92	CIC2 interrupt
End of Table 3-21		

PRODUCT PREVIEW

Table 3-22 lists the CIC2 event inputs.

Table 3-22 CIC2 Event Inputs (Secondary Events for EDMA3CC and Hyperlink) (Part 1 of 12)

Event No.	Event Name	Description
0	GPIO_INT8	GPIO interrupt
1	GPIO_INT9	GPIO interrupt
2	GPIO_INT10	GPIO interrupt
3	GPIO_INT11	GPIO interrupt
4	GPIO_INT12	GPIO interrupt

Table 3-22 CIC2 Event Inputs (Secondary Events for EDMA3CC and Hyperlink) (Part 2 of 12)

Event No.	Event Name	Description
5	GPIO_INT13	GPIO interrupt
6	GPIO_INT14	GPIO interrupt
7	GPIO_INT15	GPIO interrupt
8	DBGTBR_DMAINT	Debug trace buffer (TBR) DMA event
9	Reserved	Reserved
10	Reserved	Reserved
11	Reserved	Reserved
12	Reserved	Reserved
13	Reserved	Reserved
14	Reserved	Reserved
15	Reserved	Reserved
16	Reserved	Reserved
17	Reserved	Reserved
18	Reserved	Reserved
19	Reserved	Reserved
20	Reserved	Reserved
21	Reserved	Reserved
22	Reserved	Reserved
23	DFT_PBIST_CPU_INT	Reserved
24	QMSS_INTD_1_HIGH_16	Navigator interrupt
25	QMSS_INTD_1_HIGH_17	Navigator interrupt
26	QMSS_INTD_1_HIGH_18	Navigator interrupt
27	QMSS_INTD_1_HIGH_19	Navigator interrupt
28	QMSS_INTD_1_HIGH_20	Navigator interrupt
29	QMSS_INTD_1_HIGH_21	Navigator interrupt
30	QMSS_INTD_1_HIGH_22	Navigator interrupt
31	QMSS_INTD_1_HIGH_23	Navigator interrupt
32	QMSS_INTD_1_HIGH_24	Navigator interrupt
33	QMSS_INTD_1_HIGH_25	Navigator interrupt
34	QMSS_INTD_1_HIGH_26	Navigator interrupt
35	QMSS_INTD_1_HIGH_27	Navigator interrupt
36	QMSS_INTD_1_HIGH_28	Navigator interrupt
37	QMSS_INTD_1_HIGH_29	Navigator interrupt
38	QMSS_INTD_1_HIGH_30	Navigator interrupt
39	QMSS_INTD_1_HIGH_31	Navigator interrupt
40	NETCP_0_MDIO_LINK_INT0	Packet Accelerator 0 subsystem MDIO interrupt
41	NETCP_0_MDIO_LINK_INT1	Packet Accelerator 0 subsystem MDIO interrupt
42	NETCP_0_MDIO_USER_INT0	Packet Accelerator 0 subsystem MDIO interrupt
43	NETCP_0_MDIO_USER_INT1	Packet Accelerator 0 subsystem MDIO interrupt
44	NETCP_0_MISC_INT	Packet Accelerator 0 subsystem MDIO interrupt
45	Reserved	Reserved
46	Reserved	Reserved
47	Reserved	Reserved
48	Reserved	Reserved

Table 3-22 CIC2 Event Inputs (Secondary Events for EDMA3CC and Hyperlink) (Part 3 of 12)

Event No.	Event Name	Description
49	TRACER_DDR_INT	Tracer sliding time window interrupt for MSMC-DDR3A
50	TRACER_MSMC_0_INT	Tracer sliding time window interrupt for MSMC SRAM bank0
51	TRACER_MSMC_1_INT	Tracer sliding time window interrupt for MSMC SRAM bank1
52	TRACER_MSMC_2_INT	Tracer sliding time window interrupt for MSMC SRAM bank2
53	TRACER_MSMC_3_INT	Tracer sliding time window interrupt for MSMC SRAM bank3
54	TRACER_CFG_INT	Tracer sliding time window interrupt for TeraNet CFG
55	TRACER_QMSS_QM_CFG1_INT	Tracer sliding time window interrupt for Navigator CFG1 slave port
56	TRACER_QMSS_DMA_INT	Tracer sliding time window interrupt for Navigator VBUSM slave port
57	TRACER_SEM_INT	Tracer sliding time window interrupt for Semaphore interrupt
58	Reserved	Reserved
59	Reserved	Reserved
60	Reserved	Reserved
61	Reserved	Reserved
62	BOOTCFG_INT	BOOTCFG error interrupt
63	NETCP_0_PKTDMA_INT0	Packet Accelerator0 Packet DMA starvation interrupt
64	MPU_0_INT	MPU0 interrupt
65	MSMC_SCRUB_CERROR	MSMC error interrupt
66	MPU_1_INT	MPU1 interrupt
67	Reserved	Reserved
68	MPU_2_INT	MPU2 interrupt
69	QMSS_INTD_1_PKTDMA_0	Navigator Packet DMA interrupt
70	Reserved	Reserved
71	QMSS_INTD_1_PKTDMA_1	Navigator Packet DMA interrupt
72	MSMC_DEDC_CERROR	MSMC error interrupt
73	MSMC_DEDC_NC_ERROR	MSMC error interrupt
74	MSMC_SCRUB_NC_ERROR	MSMC error interrupt
75	Reserved	Reserved
76	MSMC_MPF_ERROR0	Memory protection fault indicators for system master PrivID = 0
77	Reserved	Reserved
78	Reserved	Reserved
79	Reserved	Reserved
80	Reserved	Reserved
81	Reserved	Reserved
82	Reserved	Reserved
83	Reserved	Reserved
84	MSMC_MPF_ERROR8	Memory protection fault indicators for system master PrivID = 8
85	MSMC_MPF_ERROR9	Memory protection fault indicators for system master PrivID = 9
86	MSMC_MPF_ERROR10	Memory protection fault indicators for system master PrivID = 10
87	MSMC_MPF_ERROR11	Memory protection fault indicators for system master PrivID = 11
88	MSMC_MPF_ERROR12	Memory protection fault indicators for system master PrivID = 12
89	MSMC_MPF_ERROR13	Memory protection fault indicators for system master PrivID = 13
90	MSMC_MPF_ERROR14	Memory protection fault indicators for system master PrivID = 14
91	MSMC_MPF_ERROR15	Memory protection fault indicators for system master PrivID = 15
92	Reserved	Reserved

Table 3-22 CIC2 Event Inputs (Secondary Events for EDMA3CC and Hyperlink) (Part 4 of 12)

Event No.	Event Name	Description
93	GPIO_INT16	GPIO interrupt
94	GPIO_INT17	GPIO interrupt
95	GPIO_INT18	GPIO interrupt
96	GPIO_INT19	GPIO interrupt
97	GPIO_INT20	GPIO interrupt
98	GPIO_INT21	GPIO interrupt
99	GPIO_INT22	GPIO interrupt
100	GPIO_INT23	GPIO interrupt
101	GPIO_INT24	GPIO interrupt
102	GPIO_INT25	GPIO interrupt
103	GPIO_INT26	GPIO interrupt
104	GPIO_INT27	GPIO interrupt
105	GPIO_INT28	GPIO interrupt
106	GPIO_INT29	GPIO interrupt
107	GPIO_INT30	GPIO interrupt
108	GPIO_INT31	GPIO interrupt
109	Reserved	Reserved
110	Reserved	Reserved
111	Reserved	Reserved
112	Reserved	Reserved
113	Reserved	Reserved
114	Reserved	Reserved
115	Reserved	Reserved
116	Reserved	Reserved
117	AEMIF_EASYNCERR	Asynchronous EMIF16 error interrupt
118	Reserved	Reserved
119	Reserved	Reserved
120	Reserved	Reserved
121	Reserved	Reserved
122	Reserved	Reserved
123	Reserved	Reserved
124	Reserved	Reserved
125	Reserved	Reserved
126	Reserved	Reserved
127	Reserved	Reserved
128	Reserved	Reserved
129	Reserved	Reserved
130	Reserved	Reserved
131	Reserved	Reserved
132	Reserved	Reserved
133	Reserved	Reserved
134	Reserved	Reserved
135	Reserved	Reserved
136	Reserved	Reserved

Table 3-22 CIC2 Event Inputs (Secondary Events for EDMA3CC and Hyperlink) (Part 5 of 12)

Event No.	Event Name	Description
137	Reserved	Reserved
138	QMSS_INTD_1_HIGH_0	Navigator hi interrupt
139	QMSS_INTD_1_HIGH_1	Navigator hi interrupt
140	QMSS_INTD_1_HIGH_2	Navigator hi interrupt
141	QMSS_INTD_1_HIGH_3	Navigator hi interrupt
142	QMSS_INTD_1_HIGH_4	Navigator hi interrupt
143	QMSS_INTD_1_HIGH_5	Navigator hi interrupt
144	QMSS_INTD_1_HIGH_6	Navigator hi interrupt
145	QMSS_INTD_1_HIGH_7	Navigator hi interrupt
146	QMSS_INTD_1_HIGH_8	Navigator hi interrupt
147	QMSS_INTD_1_HIGH_9	Navigator hi interrupt
148	QMSS_INTD_1_HIGH_10	Navigator hi interrupt
149	QMSS_INTD_1_HIGH_11	Navigator hi interrupt
150	QMSS_INTD_1_HIGH_12	Navigator hi interrupt
151	QMSS_INTD_1_HIGH_13	Navigator hi interrupt
152	QMSS_INTD_1_HIGH_14	Navigator hi interrupt
153	QMSS_INTD_1_HIGH_15	Navigator hi interrupt
154	QMSS_INTD_2_HIGH_0	Navigator second hi interrupt
155	QMSS_INTD_2_HIGH_1	Navigator second hi interrupt
156	QMSS_INTD_2_HIGH_2	Navigator second hi interrupt
157	QMSS_INTD_2_HIGH_3	Navigator second hi interrupt
158	QMSS_INTD_2_HIGH_4	Navigator second hi interrupt
159	QMSS_INTD_2_HIGH_5	Navigator second hi interrupt
160	QMSS_INTD_2_HIGH_6	Navigator second hi interrupt
161	QMSS_INTD_2_HIGH_7	Navigator second hi interrupt
162	QMSS_INTD_2_HIGH_8	Navigator second hi interrupt
163	QMSS_INTD_2_HIGH_9	Navigator second hi interrupt
164	QMSS_INTD_2_HIGH_10	Navigator second hi interrupt
165	QMSS_INTD_2_HIGH_11	Navigator second hi interrupt
166	QMSS_INTD_2_HIGH_12	Navigator second hi interrupt
167	QMSS_INTD_2_HIGH_13	Navigator second hi interrupt
168	QMSS_INTD_2_HIGH_14	Navigator second hi interrupt
169	QMSS_INTD_2_HIGH_15	Navigator second hi interrupt
170	QMSS_INTD_2_HIGH_16	Navigator second hi interrupt
171	QMSS_INTD_2_HIGH_17	Navigator second hi interrupt
172	QMSS_INTD_2_HIGH_18	Navigator second hi interrupt
173	QMSS_INTD_2_HIGH_19	Navigator second hi interrupt
174	QMSS_INTD_2_HIGH_20	Navigator second hi interrupt
175	QMSS_INTD_2_HIGH_21	Navigator second hi interrupt
176	QMSS_INTD_2_HIGH_22	Navigator second hi interrupt
177	QMSS_INTD_2_HIGH_23	Navigator second hi interrupt
178	QMSS_INTD_2_HIGH_24	Navigator second hi interrupt
179	QMSS_INTD_2_HIGH_25	Navigator second hi interrupt
180	QMSS_INTD_2_HIGH_26	Navigator second hi interrupt

PRODUCT PREVIEW

Table 3-22 CIC2 Event Inputs (Secondary Events for EDMA3CC and Hyperlink) (Part 6 of 12)

Event No.	Event Name	Description
181	QMSS_INTD_2_HIGH_27	Navigator second hi interrupt
182	QMSS_INTD_2_HIGH_28	Navigator second hi interrupt
183	QMSS_INTD_2_HIGH_29	Navigator second hi interrupt
184	QMSS_INTD_2_HIGH_30	Navigator second hi interrupt
185	QMSS_INTD_2_HIGH_31	Navigator second hi interrupt
186	MPU_12_INT	MPU12 addressing violation interrupt and protection violation interrupt
187	MPU_13_INT	MPU13 addressing violation interrupt and protection violation interrupt
188	MPU_14_INT	MPU14 addressing violation interrupt and protection violation interrupt
189	MPU_15_INT	MPU15 addressing violation interrupt and protection violation interrupt
190	Reserved	Reserved
191	Reserved	Reserved
192	Reserved	Reserved
193	Reserved	Reserved
194	Reserved	Reserved
195	Reserved	Reserved
196	Reserved	Reserved
197	Reserved	Reserved
198	Reserved	Reserved
199	TRACER_QMSS_QM_CFG2_INT	Tracer sliding time window interrupt for Navigator CFG2 slave port
200	TRACER_EDMACC_0	Tracer sliding time window interrupt foR EDMA3CC0
201	TRACER_EDMACC_123_INT	Tracer sliding time window interrupt for EDMA3CC1, EDMA3CC2, and EDMA3CC3
202	TRACER_CIC_INT	Tracer sliding time window interrupt for interrupt controllers (CIC)
203	Reserved	Reserved
204	MPU_5_INT	MPU5 addressing violation interrupt and protection violation interrupt
205	Reserved	Reserved
206	MPU_7_INT	MPU7 addressing violation interrupt and protection violation interrupt
207	MPU_8_INT	MPU8 addressing violation interrupt and protection violation interrupt
208	Reserved	Reserved
209	Reserved	Reserved
210	Reserved	Reserved
211	DDR3_0_ERR	DDR3A error interrupt
212	HYPERLINK_0_INT	HyperLink interrupt
213	EDMACC_0_ERRINT	EDMA3CC0 error interrupt
214	EDMACC_0_MPINT	EDMA3CC0 memory protection interrupt
215	EDMACC_0_TC_0_ERRINT	EDMA3CC0 TPTC0 error interrupt
216	EDMACC_0_TC_1_ERRINT	EDMA3CC0 TPTC1 error interrupt
217	EDMACC_1_ERRINT	EDMA3CC1 error interrupt
218	EDMACC_1_MPINT	EDMA3CC1 memory protection interrupt
219	EDMACC_1_TC_0_ERRINT	EDMA3CC1 TPTC0 error interrupt
220	EDMACC_1_TC_1_ERRINT	EDMA3CC1 TPTC1 error interrupt
221	EDMACC_1_TC_2_ERRINT	EDMA3CC1 TPTC2 error interrupt
222	EDMACC_1_TC_3_ERRINT	EDMA3CC1 TPTC3 error interrupt
223	EDMACC_2_ERRINT	EDMA3CC2 error interrupt
224	EDMACC_2_MPINT	EDMA3CC2 memory protection interrupt

Table 3-22 CIC2 Event Inputs (Secondary Events for EDMA3CC and Hyperlink) (Part 7 of 12)

Event No.	Event Name	Description
225	EDMACC_2_TC_0_ERRINT	EDMA3CC2 TPTC0 error interrupt
226	EDMACC_2_TC_1_ERRINT	EDMA3CC2 TPTC1 error interrupt
227	EDMACC_2_TC_2_ERRINT	EDMA3CC2 TPTC2 error interrupt
228	EDMACC_2_TC_3_ERRINT	EDMA3CC2 TPTC3 error interrupt
229	EDMACC_3_ERRINT	EDMA3CC3 error interrupt
230	EDMACC_3_MPINT	EDMA3CC3 memory protection interrupt
231	EDMACC_3_TC_0_ERRINT	EDMA3CC3 TPTC0 error interrupt
232	EDMACC_3_TC_1_ERRINT	EDMA3CC3 TPTC1 error interrupt
233	EDMACC_4_ERRINT	EDMA3CC4 error interrupt
234	EDMACC_4_MPINT	EDMA3CC4 memory protection interrupt
235	EDMACC_4_TC_0_ERRINT	EDMA3CC4 TPTC0 error interrupt
236	EDMACC_4_TC_1_ERRINT	EDMA3CC4 TPTC1 error interrupt
237	QMSS_QUE_PEND_652	Navigator transmit queue pending event for indicated queue
238	QMSS_QUE_PEND_653	Navigator transmit queue pending event for indicated queue
239	QMSS_QUE_PEND_654	Navigator transmit queue pending event for indicated queue
240	QMSS_QUE_PEND_655	Navigator transmit queue pending event for indicated queue
241	QMSS_QUE_PEND_656	Navigator transmit queue pending event for indicated queue
242	QMSS_QUE_PEND_657	Navigator transmit queue pending event for indicated queue
243	QMSS_QUE_PEND_658	Navigator transmit queue pending event for indicated queue
244	QMSS_QUE_PEND_659	Navigator transmit queue pending event for indicated queue
245	QMSS_QUE_PEND_660	Navigator transmit queue pending event for indicated queue
246	QMSS_QUE_PEND_661	Navigator transmit queue pending event for indicated queue
247	QMSS_QUE_PEND_662	Navigator transmit queue pending event for indicated queue
248	QMSS_QUE_PEND_663	Navigator transmit queue pending event for indicated queue
249	QMSS_QUE_PEND_664	Navigator transmit queue pending event for indicated queue
250	QMSS_QUE_PEND_665	Navigator transmit queue pending event for indicated queue
251	QMSS_QUE_PEND_666	Navigator transmit queue pending event for indicated queue
252	QMSS_QUE_PEND_667	Navigator transmit queue pending event for indicated queue
253	QMSS_QUE_PEND_668	Navigator transmit queue pending event for indicated queue
254	QMSS_QUE_PEND_669	Navigator transmit queue pending event for indicated queue
255	QMSS_QUE_PEND_670	Navigator transmit queue pending event for indicated queue
256	QMSS_QUE_PEND_671	Navigator transmit queue pending event for indicated queue
257	QMSS_QUE_PEND_672	Navigator transmit queue pending event for indicated queue
258	QMSS_QUE_PEND_673	Navigator transmit queue pending event for indicated queue
259	QMSS_QUE_PEND_674	Navigator transmit queue pending event for indicated queue
260	QMSS_QUE_PEND_675	Navigator transmit queue pending event for indicated queue
261	QMSS_QUE_PEND_676	Navigator transmit queue pending event for indicated queue
262	QMSS_QUE_PEND_677	Navigator transmit queue pending event for indicated queue
263	QMSS_QUE_PEND_678	Navigator transmit queue pending event for indicated queue
264	QMSS_QUE_PEND_679	Navigator transmit queue pending event for indicated queue
265	QMSS_QUE_PEND_680	Navigator transmit queue pending event for indicated queue
266	QMSS_QUE_PEND_681	Navigator transmit queue pending event for indicated queue
267	QMSS_QUE_PEND_682	Navigator transmit queue pending event for indicated queue
268	QMSS_QUE_PEND_683	Navigator transmit queue pending event for indicated queue

Table 3-22 CIC2 Event Inputs (Secondary Events for EDMA3CC and Hyperlink) (Part 8 of 12)

Event No.	Event Name	Description
269	QMSS_QUE_PEND_684	Navigator transmit queue pending event for indicated queue
270	QMSS_QUE_PEND_685	Navigator transmit queue pending event for indicated queue
271	QMSS_QUE_PEND_686	Navigator transmit queue pending event for indicated queue
272	QMSS_QUE_PEND_687	Navigator transmit queue pending event for indicated queue
273	QMSS_QUE_PEND_688	Navigator transmit queue pending event for indicated queue
274	QMSS_QUE_PEND_689	Navigator transmit queue pending event for indicated queue
275	QMSS_QUE_PEND_690	Navigator transmit queue pending event for indicated queue
276	QMSS_QUE_PEND_691	Navigator transmit queue pending event for indicated queue
277	10GbE_LINK_INT0	10 Gigabit Ethernet subsystem MDIO interrupt
278	10GbE_LINK_INT1	10 Gigabit Ethernet subsystem MDIO interrupt
279	10GbE_USER_INT0	10 Gigabit Ethernet subsystem MDIO interrupt
280	10GbE_USER_INT1	10 Gigabit Ethernet subsystem MDIO interrupt
281	10GbE_MISC_INT	10 Gigabit Ethernet subsystem MDIO interrupt
282	10GbE_INT_PKTDMA_0	10 Gigabit Ethernet Packet DMA starvation interrupt
283	Reserved	Reserved
284	Reserved	Reserved
285	Reserved	Reserved
286	Reserved	Reserved
287	Reserved	Reserved
288	Reserved	Reserved
289	Reserved	Reserved
290	Reserved	Reserved
291	SEM_INT8	Semaphore interrupt
292	SEM_INT9	Semaphore interrupt
293	SEM_INT10	Semaphore interrupt
294	SEM_INT11	Semaphore interrupt
295	SEM_INT12	Semaphore interrupt
296	Reserved	Reserved
297	Reserved	Reserved
298	Reserved	Reserved
299	SEM_ERR8	Semaphore error interrupt
300	SEM_ERR9	Semaphore error interrupt
301	SEM_ERR10	Semaphore error interrupt
302	SEM_ERR11	Semaphore error interrupt
303	SEM_ERR12	Semaphore error interrupt
304	QMSS1_ECC_INTR	Navigator ECC error interrupt
305	QMSS_INTD_1_LOW_0	Navigator interrupt
306	QMSS_INTD_1_LOW_1	Navigator interrupt
307	QMSS_INTD_1_LOW_2	Navigator interrupt
308	QMSS_INTD_1_LOW_3	Navigator interrupt
309	QMSS_INTD_1_LOW_4	Navigator interrupt
310	QMSS_INTD_1_LOW_5	Navigator interrupt
311	QMSS_INTD_1_LOW_6	Navigator interrupt
312	QMSS_INTD_1_LOW_7	Navigator interrupt

Table 3-22 CIC2 Event Inputs (Secondary Events for EDMA3CC and Hyperlink) (Part 9 of 12)

Event No.	Event Name	Description
313	QMSS_INTD_1_LOW_8	Navigator interrupt
314	QMSS_INTD_1_LOW_9	Navigator interrupt
315	QMSS_INTD_1_LOW_10	Navigator interrupt
316	QMSS_INTD_1_LOW_11	Navigator interrupt
317	QMSS_INTD_1_LOW_12	Navigator interrupt
318	QMSS_INTD_1_LOW_13	Navigator interrupt
319	QMSS_INTD_1_LOW_14	Navigator interrupt
320	QMSS_INTD_1_LOW_15	Navigator interrupt
321	QMSS_INTD_2_LOW_0	Navigator second interrupt
322	QMSS_INTD_2_LOW_1	Navigator second interrupt
323	QMSS_INTD_2_LOW_2	Navigator second interrupt
324	QMSS_INTD_2_LOW_3	Navigator second interrupt
325	QMSS_INTD_2_LOW_4	Navigator second interrupt
326	QMSS_INTD_2_LOW_5	Navigator second interrupt
327	QMSS_INTD_2_LOW_6	Navigator second interrupt
328	QMSS_INTD_2_LOW_7	Navigator second interrupt
329	QMSS_INTD_2_LOW_8	Navigator second interrupt
330	QMSS_INTD_2_LOW_9	Navigator second interrupt
331	QMSS_INTD_2_LOW_10	Navigator second interrupt
332	QMSS_INTD_2_LOW_11	Navigator second interrupt
333	QMSS_INTD_2_LOW_12	Navigator second interrupt
334	QMSS_INTD_2_LOW_13	Navigator second interrupt
335	QMSS_INTD_2_LOW_14	Navigator second interrupt
336	QMSS_INTD_2_LOW_15	Navigator interrupt
337	Reserved	Reserved
338	Reserved	Reserved
339	Reserved	Reserved
340	Reserved	Reserved
341	Reserved	Reserved
342	Reserved	Reserved
343	Reserved	Reserved
344	Reserved	Reserved
345	Reserved	Reserved
346	Reserved	Reserved
347	Reserved	Reserved
348	Reserved	Reserved
349	Reserved	Reserved
350	Reserved	Reserved
351	Reserved	Reserved
352	Reserved	Reserved
353	Reserved	Reserved
354	Reserved	Reserved
355	Reserved	Reserved
356	Reserved	Reserved

PRODUCT PREVIEW

Table 3-22 CIC2 Event Inputs (Secondary Events for EDMA3CC and Hyperlink) (Part 10 of 12)

Event No.	Event Name	Description
357	Reserved	Reserved
358	Reserved	Reserved
359	Reserved	Reserved
360	Reserved	Reserved
361	Reserved	Reserved
362	PSC_ALLINT	PSC interrupt
363	Reserved	Reserved
364	NETCP_1_MDIO_LINK_INT0	Packet Accelerator 1subsystem MDIO interrupt
365	NETCP_1_MDIO_LINK_INT1	Packet Accelerator 1subsystem MDIO interrupt
366	NETCP_1_MDIO_USER_INT0	Packet Accelerator 1subsystem MDIO interrupt
367	NETCP_1_MDIO_USER_INT1	Packet Accelerator 1subsystem MDIO interrupt
368	NETCP_1_MISC_INT	Packet Accelerator 1subsystem MDIO interrupt
369	NETCP_1_PKTDMA_INT0	Packet Accelerator1 Packet DMA starvation interrupt
370	Reserved	Reserved
371	Reserved	Reserved
372	MPU_9_INT	MPU9 addressing violation interrupt and protection violation interrupt
373	MPU_10_INT	MPU10 addressing violation interrupt and protection violation interrupt
374	MPU_11_INT	MPU11 addressing violation interrupt and protection violation interrupt
375	TRACER_MSMC_4_INT	Tracer sliding time window interrupt for MSMC SRAM bank 4
376	TRACER_MSMC_5_INT	Tracer sliding time window interrupt for MSMC SRAM bank 4
377	TRACER_MSMC_6_INT	Tracer sliding time window interrupt for MSMC SRAM bank 4
378	TRACER_MSMC_7_INT	Tracer sliding time window interrupt for MSMC SRAM bank 4
379	TRACER_PCIE1_INT	Tracer sliding time window interrupt for PCIE1
380	Reserved	Reserved
381	Reserved	Reserved
382	Reserved	Reserved
383	Reserved	Reserved
384	TRACER_SPI_ROM_EMIF_INT	Tracer sliding time window interrupt for SPI/ROM/EMIF16 modules
385	Reserved	Reserved
386	TRACER_NETCP1_INT	Tracer sliding time window interrupt for NetCP1, USB1 CFG port tracer
387	TIMER_8_INTL	Timer interrupt low
388	TIMER_8_INTH	Timer interrupt high
389	TIMER_9_INTL	Timer interrupt low
390	TIMER_9_INTH	Timer interrupt high
391	TIMER_10_INTL	Timer interrupt low
392	TIMER_10_INTH	Timer interrupt high
393	TIMER_11_INTL	Timer interrupt low
394	TIMER_11_INTH	Timer interrupt high
395	TIMER_14_INTL	Timer interrupt low
396	TIMER_14_INTH	Timer interrupt high
397	TIMER_15_INTL	Timer interrupt low
398	TIMER_15_INTH	Timer interrupt high
399	USB_0_INT00	USB 0 event ring 0 interrupt
400	USB_0_INT01	USB 0 event ring 1 interrupt

Table 3-22 CIC2 Event Inputs (Secondary Events for EDMA3CC and Hyperlink) (Part 11 of 12)

Event No.	Event Name	Description
401	USB_0_INT02	USB 0 event ring 2 interrupt
402	USB_0_INT03	USB 0 event ring 3 interrupt
403	USB_0_INT04	USB 0 event ring 4 interrupt
404	USB_0_INT05	USB 0 event ring 5 interrupt
405	USB_0_INT06	USB 0 event ring 6 interrupt
406	USB_0_INT07	USB 0 event ring 7 interrupt
407	USB_0_INT08	USB 0 event ring 8 interrupt
408	USB_0_INT09	USB 0 event ring 9 interrupt
409	USB_0_INT10	USB 0 event ring 10 interrupt
410	USB_0_INT11	USB 0 event ring 11 interrupt
411	USB_0_INT12	USB 0 event ring 12 interrupt
412	USB_0_INT13	USB 0 event ring 13 interrupt
413	USB_0_INT14	USB 0 event ring 14 interrupt
414	USB_0_INT15	USB 0 event ring 15 interrupt
415	USB_0_MISCINT	USB 0 Miscellaneous interrupt
416	USB_0_OABSINT	USB 0 OABS interrupt
417	Reserved	Reserved
418	USB_1_INT00	USB 1 event ring 0 interrupt
419	USB_1_INT01	USB 1 event ring 1 interrupt
420	USB_1_INT02	USB 1 event ring 2 interrupt
421	USB_1_INT03	USB 1 event ring 3 interrupt
422	USB_1_INT04	USB 1 event ring 4 interrupt
423	USB_1_INT05	USB 1 event ring 5 interrupt
424	USB_1_INT06	USB 1 event ring 6 interrupt
425	USB_1_INT07	USB 1 event ring 7 interrupt
426	USB_1_INT08	USB 1 event ring 8 interrupt
427	USB_1_INT09	USB 1 event ring 9 interrupt
428	USB_1_INT10	USB 1 event ring 10 interrupt
429	USB_1_INT11	USB 1 event ring 11 interrupt
430	USB_1_INT12	USB 1 event ring 12 interrupt
431	USB_1_INT13	USB 1 event ring 13 interrupt
432	USB_1_INT14	USB 1 event ring 14 interrupt
433	USB_1_INT15	USB 1 event ring 15 interrupt
434	USB_1_MISCINT	USB 1 miscellaneous interrupt
435	USB_1_OABSINT	USB 1 OABS interrupt
436	Reserved	Reserved
437	Reserved	Reserved
438	Reserved	Reserved
439	Reserved	Reserved
440	Reserved	Reserved
441	Reserved	Reserved
442	Reserved	Reserved
443	Reserved	Reserved
444	Reserved	Reserved

Table 3-22 CIC2 Event Inputs (Secondary Events for EDMA3CC and Hyperlink) (Part 12 of 12)

Event No.	Event Name	Description
445	Reserved	Reserved
446	TIMER_12_INTL	Timer interrupt low
447	TIMER_12_INTH	Timer interrupt high
448	TIMER_13_INTL	Timer interrupt low
449	TIMER_13_INTH	Timer interrupt high
450	TIMER_16_INTL	Timer interrupt low
451	TIMER_16_INTH	Timer interrupt high
452	TIMER_17_INTL	Timer interrupt low
453	TIMER_17_INTH	Timer interrupt high
454	TIMER_18_INTL	Timer interrupt low
455	TIMER_18_INTH	Timer interrupt high
456	TIMER_19_INTL	Timer interrupt low
457	TIMER_19_INTH	Timer interrupt high
458	Reserved	Reserved
459	RSTMUX_INT8	Boot config watchdog timer expiration event for ARM Core 0
460	RSTMUX_INT9	Boot config watchdog timer expiration event for ARM Core 1
461	RSTMUX_INT10	Boot config watchdog timer expiration event for ARM Core 2
462	RSTMUX_INT11	Boot config watchdog timer expiration event for ARM Core 3
463	GPIO_INT0	GPIO interrupt
464	GPIO_INT1	GPIO interrupt
465	GPIO_INT2	GPIO interrupt
466	GPIO_INT3	GPIO interrupt
467	GPIO_INT4	GPIO interrupt
468	GPIO_INT5	GPIO interrupt
469	GPIO_INT6	GPIO interrupt
470	GPIO_INT7	GPIO interrupt
471	Reserved	Reserved
472	Reserved	Reserved
473	Reserved	Reserved
474	Reserved	Reserved
475	Reserved	Reserved
476	Reserved	Reserved
477	Reserved	Reserved
478	Reserved	Reserved
End of Table 3-22		

3.3.2 CIC Registers

This section includes the CIC memory map information and registers.

3.3.2.1 CIC0 Register Map

Table 3-23 CIC0 Registers (Part 1 of 3)

Address Offset	Register Mnemonic	Register Name
0x0	REVISION_REG	Revision Register
0x4	CONTROL_REG	Control Register
0xc	HOST_CONTROL_REG	Host Control Register
0x10	GLOBAL_ENABLE_HINT_REG	Global Host Int Enable Register
0x20	STATUS_SET_INDEX_REG	Status Set Index Register
0x24	STATUS_CLR_INDEX_REG	Status Clear Index Register
0x28	ENABLE_SET_INDEX_REG	Enable Set Index Register
0x2c	ENABLE_CLR_INDEX_REG	Enable Clear Index Register
0x34	HINT_ENABLE_SET_INDEX_REG	Host Int Enable Set Index Register
0x38	HINT_ENABLE_CLR_INDEX_REG	Host Int Enable Clear Index Register
0x200	RAW_STATUS_REG0	Raw Status Register 0
0x204	RAW_STATUS_REG1	Raw Status Register 1
0x208	RAW_STATUS_REG2	Raw Status Register 2
0x20c	RAW_STATUS_REG3	Raw Status Register 3
0x210	RAW_STATUS_REG4	Raw Status Register 4
0x214	RAW_STATUS_REG5	Raw Status Register 5
0x218	RAW_STATUS_REG6	Raw Status Register 6
0x280	ENA_STATUS_REG0	Enabled Status Register 0
0x284	ENA_STATUS_REG1	Enabled Status Register 1
0x288	ENA_STATUS_REG2	Enabled Status Register 2
0x28c	ENA_STATUS_REG3	Enabled Status Register 3
0x290	ENA_STATUS_REG4	Enabled Status Register 4
0x294	ENA_STATUS_REG5	Enabled Status Register 5
0x298	ENA_STATUS_REG6	Enabled Status Register 6
0x300	ENABLE_REG0	Enable Register 0
0x304	ENABLE_REG1	Enable Register 1
0x308	ENABLE_REG2	Enable Register 2
0x30c	ENABLE_REG3	Enable Register 3
0x310	ENABLE_REG4	Enable Register 4
0x314	ENABLE_REG5	Enable Register 5
0x318	ENABLE_REG6	Enable Register 6
0x380	ENABLE_CLR_REG0	Enable Clear Register 0
0x384	ENABLE_CLR_REG1	Enable Clear Register 1
0x388	ENABLE_CLR_REG2	Enable Clear Register 2
0x38c	ENABLE_CLR_REG3	Enable Clear Register 3
0x390	ENABLE_CLR_REG4	Enable Clear Register 4
0x394	ENABLE_CLR_REG5	Enable Clear Register 5
0x398	ENABLE_CLR_REG6	Enable Clear Register 6
0x400	CH_MAP_REG0	Interrupt Channel Map Register for 0 to 0+3

Table 3-23 CIC0 Registers (Part 2 of 3)

Address Offset	Register Mnemonic	Register Name
0x404	CH_MAP_REG1	Interrupt Channel Map Register for 4 to 4+3
0x408	CH_MAP_REG2	Interrupt Channel Map Register for 8 to 8+3
0x40c	CH_MAP_REG3	Interrupt Channel Map Register for 12 to 12+3
0x410	CH_MAP_REG4	Interrupt Channel Map Register for 16 to 16+3
0x414	CH_MAP_REG5	Interrupt Channel Map Register for 20 to 20+3
0x418	CH_MAP_REG6	Interrupt Channel Map Register for 24 to 24+3
0x41c	CH_MAP_REG7	Interrupt Channel Map Register for 28 to 28+3
0x420	CH_MAP_REG8	Interrupt Channel Map Register for 32 to 32+3
0x424	CH_MAP_REG9	Interrupt Channel Map Register for 36 to 36+3
0x428	CH_MAP_REG10	Interrupt Channel Map Register for 40 to 40+3
0x42c	CH_MAP_REG11	Interrupt Channel Map Register for 44 to 44+3
0x430	CH_MAP_REG12	Interrupt Channel Map Register for 48 to 48+3
0x434	CH_MAP_REG13	Interrupt Channel Map Register for 52 to 52+3
0x438	CH_MAP_REG14	Interrupt Channel Map Register for 56 to 56+3
0x43c	CH_MAP_REG15	Interrupt Channel Map Register for 60 to 60+3
0x440	CH_MAP_REG16	Interrupt Channel Map Register for 64 to 64+3
0x444	CH_MAP_REG17	Interrupt Channel Map Register for 68 to 68+3
0x448	CH_MAP_REG18	Interrupt Channel Map Register for 72 to 72+3
0x44c	CH_MAP_REG19	Interrupt Channel Map Register for 76 to 76+3
0x450	CH_MAP_REG20	Interrupt Channel Map Register for 80 to 80+3
0x454	CH_MAP_REG21	Interrupt Channel Map Register for 84 to 84+3
0x458	CH_MAP_REG22	Interrupt Channel Map Register for 88 to 88+3
0x45c	CH_MAP_REG23	Interrupt Channel Map Register for 92 to 92+3
0x460	CH_MAP_REG24	Interrupt Channel Map Register for 96 to 96+3
0x464	CH_MAP_REG25	Interrupt Channel Map Register for 100 to 100+3
0x468	CH_MAP_REG26	Interrupt Channel Map Register for 104 to 104+3
0x46c	CH_MAP_REG27	Interrupt Channel Map Register for 108 to 108+3
0x470	CH_MAP_REG28	Interrupt Channel Map Register for 112 to 112+3
0x474	CH_MAP_REG29	Interrupt Channel Map Register for 116 to 116+3
0x478	CH_MAP_REG30	Interrupt Channel Map Register for 120 to 120+3
0x47c	CH_MAP_REG31	Interrupt Channel Map Register for 124 to 124+3
0x480	CH_MAP_REG32	Interrupt Channel Map Register for 128 to 128+3
0x484	CH_MAP_REG33	Interrupt Channel Map Register for 132 to 132+3
0x488	CH_MAP_REG34	Interrupt Channel Map Register for 136 to 136+3
0x48c	CH_MAP_REG35	Interrupt Channel Map Register for 140 to 140+3
0x490	CH_MAP_REG36	Interrupt Channel Map Register for 144 to 144+3
0x494	CH_MAP_REG37	Interrupt Channel Map Register for 148 to 148+3
0x498	CH_MAP_REG38	Interrupt Channel Map Register for 152 to 152+3
0x49c	CH_MAP_REG39	Interrupt Channel Map Register for 156 to 156+3
0x4a0	CH_MAP_REG40	Interrupt Channel Map Register for 160 to 160+3
0x4a4	CH_MAP_REG41	Interrupt Channel Map Register for 164 to 164+3
0x4a8	CH_MAP_REG42	Interrupt Channel Map Register for 168 to 168+3
0x4ac	CH_MAP_REG43	Interrupt Channel Map Register for 172 to 172+3
0x4b0	CH_MAP_REG44	Interrupt Channel Map Register for 176 to 176+3

Table 3-23 CIC0 Registers (Part 3 of 3)

Address Offset	Register Mnemonic	Register Name
0x4b4	CH_MAP_REG45	Interrupt Channel Map Register for 180 to 180+3
0x4b8	CH_MAP_REG46	Interrupt Channel Map Register for 184 to 184+3
0x4bc	CH_MAP_REG47	Interrupt Channel Map Register for 188 to 188+3
0x4c0	CH_MAP_REG48	Interrupt Channel Map Register for 192 to 192+3
0x4c4	CH_MAP_REG49	Interrupt Channel Map Register for 196 to 196+3
0x4c8	CH_MAP_REG50	Interrupt Channel Map Register for 200 to 200+3
0x4cc	CH_MAP_REG51	Interrupt Channel Map Register for 204 to 204+3
0x800	HINT_MAP_REG0	Host Interrupt Map Register for 0 to 0+3
0x804	HINT_MAP_REG1	Host Interrupt Map Register for 4 to 4+3
0x808	HINT_MAP_REG2	Host Interrupt Map Register for 8 to 8+3
0x80c	HINT_MAP_REG3	Host Interrupt Map Register for 12 to 12+3
0x810	HINT_MAP_REG4	Host Interrupt Map Register for 16 to 16+3
0x814	HINT_MAP_REG5	Host Interrupt Map Register for 20 to 20+3
0x818	HINT_MAP_REG6	Host Interrupt Map Register for 24 to 24+3
0x81c	HINT_MAP_REG7	Host Interrupt Map Register for 28 to 28+3
0x820	HINT_MAP_REG8	Host Interrupt Map Register for 32 to 32+3
0x824	HINT_MAP_REG9	Host Interrupt Map Register for 36 to 36+3
0x828	HINT_MAP_REG10	Host Interrupt Map Register for 40 to 40+3
0x82c	HINT_MAP_REG11	Host Interrupt Map Register for 44 to 44+3
0x830	HINT_MAP_REG12	Host Interrupt Map Register for 48 to 48+3
0x834	HINT_MAP_REG13	Host Interrupt Map Register for 52 to 52+3
0x838	HINT_MAP_REG14	Host Interrupt Map Register for 56 to 56+3
0x83c	HINT_MAP_REG15	Host Interrupt Map Register for 60 to 60+3
0x840	HINT_MAP_REG16	Host Interrupt Map Register for 64 to 64+3
0x844	HINT_MAP_REG17	Host Interrupt Map Register for 68 to 68+3
0x848	HINT_MAP_REG18	Host Interrupt Map Register for 72 to 72+3
0x84c	HINT_MAP_REG19	Host Interrupt Map Register for 76 to 76+3
0x1500	ENABLE_HINT_REG0	Host Int Enable Register 0
0x1504	ENABLE_HINT_REG1	Host Int Enable Register 1
0x1508	ENABLE_HINT_REG2	Host Int Enable Register 2
End of Table 3-23		

PRODUCT PREVIEW

3.3.2.2 CIC2 Register Map

Table 3-24 CIC2 Registers (Part 1 of 2)

Address Offset	Register Mnemonic	Register Name
0x0	REVISION_REG	Revision Register
0x10	GLOBAL_ENABLE_HINT_REG	Global Host Int Enable Register
0x20	STATUS_SET_INDEX_REG	Status Set Index Register
0x24	STATUS_CLR_INDEX_REG	Status Clear Index Register
0x28	ENABLE_SET_INDEX_REG	Enable Set Index Register
0x2c	ENABLE_CLR_INDEX_REG	Enable Clear Index Register
0x34	HINT_ENABLE_SET_INDEX_REG	Host Int Enable Set Index Register
0x38	HINT_ENABLE_CLR_INDEX_REG	Host Int Enable Clear Index Register

Table 3-24 CIC2 Registers (Part 2 of 2)

Address Offset	Register Mnemonic	Register Name
0x200	RAW_STATUS_REG0	Raw Status Register 0
0x204	RAW_STATUS_REG1	Raw Status Register 1
0x280	ENA_STATUS_REG0	Enabled Status Register 0
0x284	ENA_STATUS_REG1	Enabled Status Register 1
0x300	ENABLE_REG0	Enable Register 0
0x304	ENABLE_REG1	Enable Register 1
0x380	ENABLE_CLR_REG0	Enable Clear Register 0
0x384	ENABLE_CLR_REG1	Enable Clear Register 1
0x400	CH_MAP_REG0	Interrupt Channel Map Register for 0 to 0+3
0x404	CH_MAP_REG1	Interrupt Channel Map Register for 4 to 4+3
0x408	CH_MAP_REG2	Interrupt Channel Map Register for 8 to 8+3
0x40c	CH_MAP_REG3	Interrupt Channel Map Register for 12 to 12+3
0x410	CH_MAP_REG4	Interrupt Channel Map Register for 16 to 16+3
0x414	CH_MAP_REG5	Interrupt Channel Map Register for 20 to 20+3
0x418	CH_MAP_REG6	Interrupt Channel Map Register for 24 to 24+3
0x41c	CH_MAP_REG7	Interrupt Channel Map Register for 28 to 28+3
0x420	CH_MAP_REG8	Interrupt Channel Map Register for 32 to 32+3
0x424	CH_MAP_REG9	Interrupt Channel Map Register for 36 to 36+3
0x428	CH_MAP_REG10	Interrupt Channel Map Register for 40 to 40+3
0x42c	CH_MAP_REG11	Interrupt Channel Map Register for 44 to 44+3
0x430	CH_MAP_REG12	Interrupt Channel Map Register for 48 to 48+3
0x434	CH_MAP_REG13	Interrupt Channel Map Register for 52 to 52+3
0x438	CH_MAP_REG14	Interrupt Channel Map Register for 56 to 56+3
0x43c	CH_MAP_REG15	Interrupt Channel Map Register for 60 to 60+3
0x800	HINT_MAP_REG0	Host Interrupt Map Register for 0 to 0+3
0x804	HINT_MAP_REG1	Host Interrupt Map Register for 4 to 4+3
0x808	HINT_MAP_REG2	Host Interrupt Map Register for 8 to 8+3
0x80c	HINT_MAP_REG3	Host Interrupt Map Register for 12 to 12+3
0x810	HINT_MAP_REG4	Host Interrupt Map Register for 16 to 16+3
0x814	HINT_MAP_REG5	Host Interrupt Map Register for 20 to 20+3
0x818	HINT_MAP_REG6	Host Interrupt Map Register for 24 to 24+3
0x81c	HINT_MAP_REG7	Host Interrupt Map Register for 28 to 28+3
0x820	HINT_MAP_REG8	Host Interrupt Map Register for 32 to 32+3
0x824	HINT_MAP_REG9	Host Interrupt Map Register for 36 to 36+3
0x828	HINT_MAP_REG10	Host Interrupt Map Register for 40 to 40+3
0x1500	ENABLE_HINT_REG0	Host Int Enable Register 0
0x1504	ENABLE_HINT_REG1	Host Int Enable Register 1
End of Table 3-24		

3.4 Enhanced Direct Memory Access (EDMA3) Controller for AM5K2E04/02

The primary purpose of the EDMA3 is to service user-programmed data transfers between two memory-mapped slave endpoints on the device. The EDMA3 services software-driven paging transfers (e.g., data movement between external memory and internal memory), performs sorting or subframe extraction of various data structures, services event driven peripherals, and offloads data transfers from the device ARM CorePac.

There are 5 EDMA channel controllers on the device:

- EDMA3CC0 has two transfer controllers: TPTC0 and TPTC1
- EDMA3CC1 has four transfer controllers: TPTC0, TPTC1, TPTC2, and TPTC3
- EDMA3CC2 has four transfer controllers: TPTC0, TPTC1, TPTC2, and TPTC3
- EDMA3CC3 has two transfer controllers: TPTC0 and TPTC1
- EDMA3CC4 has two transfer controllers: TPTC0 and TPTC1

In the context of this document, TPTCx is associated with EDMA3CCy, and is referred to as EDMA3CCy TPTCx. Each of the transfer controllers has a direct connection to the switch fabric. Section 6.2 "Switch Fabric Connections Matrix" on page 173 lists the peripherals that can be accessed by the transfer controllers.

EDMA3CC0 is optimized to be used for transfers to/from/within the MSMC and DDR3A subsytems. The others are used for the remaining traffic.

Each EDMA3 channel controller includes the following features:

- Fully orthogonal transfer description
  - 3 transfer dimensions:
    - Array (multiple bytes)
    - Frame (multiple arrays)
    - Block (multiple frames)
  - Single event can trigger transfer of array, frame, or entire block
  - Independent indexes on source and destination
- Flexible transfer definition:
  - Increment or FIFO transfer addressing modes
  - Linking mechanism allows for ping-pong buffering, circular buffering, and repetitive/continuous transfers, all with no CPU intervention
  - Chaining allows multiple transfers to execute with one event
- 512 PaRAM entries for all EDMA3CC
  - Used to define transfer context for channels
  - Each PaRAM entry can be used as a DMA entry, QDMA entry, or link entry
- 64 DMA channels for all EDMA3CC
  - Manually triggered (CPU writes to channel controller register)
  - External event triggered
  - Chain triggered (completion of one transfer triggers another)
- 8 Quick DMA (QDMA) channels per EDMA3CCx
  - Used for software-driven transfers
  - Triggered upon writing to a single PaRAM set entry
- Two transfer controllers and two event queues with programmable system-level priority for EDMA3CC0, EDMA3CC3 and EDMA3CC4

- Four transfer controllers and four event queues with programmable system-level priority for each of EDMA3CC1 and EDMA3CC2
- Interrupt generation for transfer completion and error conditions
- Debug visibility
  - Queue watermarking/threshold allows detection of maximum usage of event queues
  - Error and status recording to facilitate debug

3.4.1 EDMA3 Device-Specific Information

The EDMA supports two addressing modes: constant addressing and increment addressing mode. Constant addressing mode is applicable to a very limited set of use cases. For most applications increment mode can be used. On the AM5K2E04/02 SoC, the EDMA can use constant addressing mode only with the enhanced Viterbi decoder coprocessor (VCP) and the enhanced turbo decoder coprocessor (TCP). Constant addressing mode is not supported by any other peripheral or internal memory in the DSP. Note that increment mode is supported by all peripherals, including VCP and TCP. For more information on these two addressing modes, see the *Enhanced Direct Memory Access 3 (EDMA3) for KeyStone Devices User Guide* in 1.8 "Related Documentation from Texas Instruments" on page 13.

For the range of memory addresses that includes EDMA3 channel controller (EDMA3CC) control registers and EDMA3 transfer controller (TPTC) control registers, see Section 3.1 "Memory Map Summary for AM5K2E04/02" on page 19. For memory offsets and other details on EDMA3CC and TPTC Control Register entries, see the *Enhanced Direct Memory Access 3 (EDMA3) for KeyStone Devices User Guide* in 1.8 "Related Documentation from Texas Instruments" on page 13.

3.4.2 EDMA3 Channel Controller Configuration

Table 3-25 shows the configuration for each of the EDMA3 channel controllers present on the device.

Table 3-25 EDMA3 Channel Controller Configuration

Description	EDMA3 CC0	EDMA3 CC1	EDMA3 CC2	EDMA3 CC3	EDMA3 CC4
Number of DMA channels in channel controller	64	64	64	64	64
Number of QDMA channels	8	8	8	8	8
Number of interrupt channels	64	64	64	64	64
Number of PaRAM set entries	512	512	512	512	512
Number of event queues	2	4	4	2	2
Number of transfer controllers	2	4	4	2	2
Memory protection existence	Yes	Yes	Yes	Yes	Yes
Number of memory protection and shadow regions	8	8	8	8	8
End of Table 3-25					

3.4.3 EDMA3 Transfer Controller Configuration

Each transfer controller on the device is designed differently based on considerations like performance requirements, system topology (like main TeraNet bus width, external memory bus width), etc. The parameters that determine the transfer controller configurations are:

- **FIFOSIZE:** Determines the size in bytes for the data FIFO that is the temporary buffer for the in-flight data. The data FIFO is where the read return data read by the TC read controller from the source endpoint is stored and subsequently written out to the destination endpoint by the TC write controller.
- **BUSWIDTH:** The width of the read and write data buses in bytes, for the TC read and write controller, respectively. This is typically equal to the bus width of the main TeraNet interface.

- **Default Burst Size (DBS):** The DBS is the maximum number of bytes per read/write command issued by a transfer controller.
- **DSTREGDEPTH:** This determines the number of destination FIFO register sets. The number of destination FIFO register sets for a transfer controller determines the maximum number of outstanding transfer requests.

All four parameters listed above are fixed by the design of the device.

Table 3-26 shows the configuration of each of the EDMA3 transfer controllers present on the device.

Table 3-26 EDMA3 Transfer Controller Configuration

Parameter	EDMA3 CC0/CC4		EDMA3 CC1				EDMA3 CC2				EDMA3CC3	
	TC0	TC1	TC0	TC1	TC2	TC3	TC0	TC1	TC2	TC3	TC0	TC1
FIFOSIZE	1024 bytes	1024 bytes	1024 bytes	1024 bytes	1024 bytes	1024 bytes	1024 bytes	1024 bytes	1024 bytes	1024 bytes	1024 bytes	1024 bytes
BUSWIDTH	32 bytes	32 bytes	16 bytes	16 bytes	16 bytes	16 bytes	16 bytes	16 bytes	16 bytes	16 bytes	16 bytes	16 bytes
DSTREGDEPTH	4 entries	4 entries	4 entries	4 entries	4 entries	4 entries	4 entries	4 entries	4 entries	4 entries	4 entries	4 entries
DBS	128 bytes	128 bytes	128 bytes	128 bytes	128 bytes	128 bytes	128 bytes	128 bytes	128 bytes	128 bytes	128 bytes	128 bytes
End of Table 3-26												

3.4.4 EDMA3 Channel Synchronization Events

The EDMA3 supports up to 64 DMA channels for all EDMA3CC that can be used to service system peripherals and to move data between system memories. DMA channels can be triggered by synchronization events generated by system peripherals. The following tables list the source of the synchronization event associated with each of the EDMA EDMA3CC DMA channels. On the AM5K2E04/02, the association of each synchronization event and DMA channel is fixed and cannot be reprogrammed.

For more detailed information on the EDMA3 module and how EDMA3 events are enabled, captured, processed, prioritized, linked, chained, and cleared, etc., see the *Enhanced Direct Memory Access 3 (EDMA3) for KeyStone Devices User Guide* in 1.8 "Related Documentation from Texas Instruments" on page 13.

Table 3-27 EDMA3CC0 Events for AM5K2E04/02 (Part 1 of 3)

Event No.	Event Name	Description
0	TIMER_8_INTL	Timer interrupt low
1	TIMER_8_INTH	Timer interrupt high
2	TIMER_9_INTL	Timer interrupt low
3	TIMER_9_INTH	Timer interrupt high
4	TIMER_10_INTL	Timer interrupt low
5	TIMER_10_INTH	Timer interrupt high
6	TIMER_11_INTL	Timer interrupt low
7	TIMER_11_INTH	Timer interrupt high
8	CIC_2_OUT66	CIC2 Interrupt Controller output
9	CIC_2_OUT67	CIC2 Interrupt Controller output
10	CIC_2_OUT68	CIC2 Interrupt Controller output
11	CIC_2_OUT69	CIC2 Interrupt Controller output
12	CIC_2_OUT70	CIC2 Interrupt Controller output
13	CIC_2_OUT71	CIC2 Interrupt Controller output
14	CIC_2_OUT72	CIC2 Interrupt Controller output
15	CIC_2_OUT73	CIC2 Interrupt Controller output
16	GPIO_INT8	GPIO interrupt

PRODUCT PREVIEW

Table 3-27 EDMA3CC0 Events for AM5K2E04/02 (Part 2 of 3)

Event No.	Event Name	Description
17	GPIO_INT9	GPIO interrupt
18	GPIO_INT10	GPIO interrupt
19	GPIO_INT11	GPIO interrupt
20	GPIO_INT12	GPIO interrupt
21	GPIO_INT13	GPIO interrupt
22	GPIO_INT14	GPIO interrupt
23	GPIO_INT15	GPIO interrupt
24	TIMER_16_INTL	Timer interrupt low
25	TIMER_16_INTH	Timer interrupt high
26	TIMER_17_INTL	Timer interrupt low
27	TIMER_17_INTH	Timer interrupt high
28	TIMER_18_INTL	Timer interrupt low
29	TIMER_18_INTH	Timer interrupt high
30	TIMER_19_INTL	Timer interrupt low
31	TIMER_19_INTH	Timer interrupt high
32	GPIO_INT0	GPIO interrupt
33	GPIO_INT1	GPIO interrupt
34	GPIO_INT2	GPIO interrupt
35	GPIO_INT3	GPIO interrupt
36	GPIO_INT4	GPIO interrupt
37	GPIO_INT5	GPIO interrupt
38	GPIO_INT6	GPIO interrupt
39	GPIO_INT7	GPIO interrupt
40	Reserved	Reserved
41	Reserved	Reserved
42	TIMER_12_INTL	Timer interrupt low
43	TIMER_12_INTH	Timer interrupt high
44	TIMER_13_INTL	Timer interrupt low
45	TIMER_13_INTH	Timer interrupt high
46	Reserved	Reserved
47	SEM_INT8	Semaphore interrupt
48	SEM_INT9	Semaphore interrupt
49	SEM_INT10	Semaphore interrupt
50	SEM_INT11	Semaphore interrupt
51	SEM_INT12	Semaphore interrupt
52	DBGTBR_DMAINT	Debug trace buffer (TBR) DMA event
53	ARM_TBR_DMA	ARM trace buffer (TBR) DMA event
54	QMSS_QUE_PEND_560	Navigator transmit queue pending event for indicated queue
55	QMSS_QUE_PEND_561	Navigator transmit queue pending event for indicated queue
56	QMSS_QUE_PEND_562	Navigator transmit queue pending event for indicated queue
57	QMSS_QUE_PEND_563	Navigator transmit queue pending event for indicated queue
58	QMSS_QUE_PEND_564	Navigator transmit queue pending event for indicated queue
59	QMSS_QUE_PEND_565	Navigator transmit queue pending event for indicated queue
60	QMSS_QUE_PEND_566	Navigator transmit queue pending event for indicated queue

Table 3-27 EDMA3CC0 Events for AM5K2E04/02 (Part 3 of 3)

Event No.	Event Name	Description
61	QMSS_QUE_PEND_567	Navigator transmit queue pending event for indicated queue
62	QMSS_QUE_PEND_568	Navigator transmit queue pending event for indicated queue
63	QMSS_QUE_PEND_569	Navigator transmit queue pending event for indicated queue
End of Table 3-27		

Table 3-28 EDMA3CC1 Events for AM5K2E04/02 (Part 1 of 2)

Event No.	Event Name	Description
0	GPIO_INT28	GPIO interrupt
1	GPIO_INT29	GPIO interrupt
2	SPI_0_XEVT	SPI0 transmit event
3	SPI_0_REVT	SPI0 receive event
4	SEM_INT8	Semaphore interrupt
5	SEM_INT9	Semaphore interrupt
6	GPIO_INT0	GPIO interrupt
7	GPIO_INT1	GPIO interrupt
8	GPIO_INT2	GPIO interrupt
9	GPIO_INT3	GPIO interrupt
10	QMSS_QUE_PEND_570	Navigator transmit queue pending event for indicated queue
11	QMSS_QUE_PEND_571	Navigator transmit queue pending event for indicated queue
12	QMSS_QUE_PEND_572	Navigator transmit queue pending event for indicated queue
13	QMSS_QUE_PEND_573	Navigator transmit queue pending event for indicated queue
14	Reserved	Reserved
15	QMSS_QUE_PEND_574	Navigator transmit queue pending event for indicated queue
16	QMSS_QUE_PEND_575	Navigator transmit queue pending event for indicated queue
17	QMSS_QUE_PEND_576	Navigator transmit queue pending event for indicated queue
18	QMSS_QUE_PEND_577	Navigator transmit queue pending event for indicated queue
19	QMSS_QUE_PEND_578	Navigator transmit queue pending event for indicated queue
20	QMSS_QUE_PEND_579	Navigator transmit queue pending event for indicated queue
21	QMSS_QUE_PEND_580	Navigator transmit queue pending event for indicated queue
22	TIMER_8_INTL	Timer interrupt low
23	TIMER_8_INTH	Timer interrupt high
24	TIMER_9_INTL	Timer interrupt low
25	TIMER_9_INTH	Timer interrupt high
26	TIMER_10_INTL	Timer interrupt low
27	TIMER_10_INTH	Timer interrupt high
28	TIMER_11_INTL	Timer interrupt low
29	TIMER_11_INTH	Timer interrupt high
30	TIMER_12_INTL	Timer interrupt low
31	TIMER_12_INTH	Timer interrupt high
32	TIMER_13_INTL	Timer interrupt low
33	TIMER_13_INTH	Timer interrupt high
34	TIMER_14_INTL	Timer interrupt low
35	TIMER_14_INTH	Timer interrupt high
36	TIMER_15_INTL	Timer interrupt low

PRODUCT PREVIEW

Table 3-28 EDMA3CC1 Events for AM5K2E04/02 (Part 2 of 2)

Event No.	Event Name	Description
37	TIMER_15_INTH	Timer interrupt high
38	SEM_INT10	Semaphore interrupt
39	SEM_INT11	Semaphore interrupt
40	SEM_INT12	Semaphore interrupt
41	SR_0_SR_TEMPSENSOR	SmartReflex temperature threshold crossing interrupt
42	TSIP_RCV_FINT0	TSIP receive frame interrupt for Channel 0
43	TSIP_XMT_FINT0	TSIP transmit frame interrupt for Channel 0
44	TSIP_RCV_SFINT0	TSIP receive super frame interrupt for Channel 0
45	TSIP_XMT_SFINT0	TSIP transmit super frame interrupt for Channel 0
46	TSIP_RCV_FINT0	TSIP receive frame interrupt for Channel 1
47	TSIP_XMT_FINT0	TSIP transmit frame interrupt for Channel 1
48	TSIP_RCV_SFINT0	TSIP receive super frame interrupt for Channel 1
49	TSIP_XMT_SFINT0	TSIP transmit super frame interrupt for Channel 1
50	CIC_2_OUT8	CIC2 Interrupt Controller output
51	GPIO_INT30	GPIO interrupt
52	GPIO_INT31	GPIO interrupt
53	I2C_0_REVT	I2C0 receive
54	I2C_0_XEVT	I2C0 transmit
55	CIC_2_OUT13	CIC2 Interrupt Controller output
56	CIC_2_OUT14	CIC2 Interrupt Controller output
57	CIC_2_OUT15	CIC2 Interrupt Controller output
58	CIC_2_OUT16	CIC2 Interrupt Controller output
59	CIC_2_OUT17	CIC2 Interrupt Controller output
60	CIC_2_OUT18	CIC2 Interrupt Controller output
61	CIC_2_OUT19	CIC2 Interrupt Controller output
62	USIM_RREQ	USIM read DMA event
63	USIM_WREQ	USIM write DMA event
End of Table 3-28		

Table 3-29 EDMA3CC2 Events for AM5K2E04/02 (Part 1 of 3)

Event No.	Event Name	Description
0	UART_1_URXEVT	UART1 receive event
1	UART_1_UTXEVT	UART1 transmit event
2	SPI_1_XEVT	SPI1 receive event
3	SPI_1_REVT	SPI1 transmit event
4	SPI_2_XEVT	SPI2 receive event
5	SPI_2_REVT	SPI2 transmit event
6	DBGTBR_DMAINT	Debug trace buffer (TBR) DMA event
7	ARM_TBR_DMA	ARM trace buffer (TBR) DMA event
8	Reserved	Reserved
9	Reserved	Reserved
10	I2C_1_REVT	I2C1 receive
11	I2C_1_XEVT	I2C1 transmit
12	I2C_2_REVT	I2C2 receive

Table 3-29 EDMA3CC2 Events for AM5K2E04/02 (Part 2 of 3)

Event No.	Event Name	Description
13	I2C_2_XEVT	I2C2 transmit
14	GPIO_INT16	GPIO interrupt
15	GPIO_INT17	GPIO interrupt
16	GPIO_INT18	GPIO interrupt
17	GPIO_INT19	GPIO interrupt
18	GPIO_INT20	GPIO interrupt
19	GPIO_INT21	GPIO interrupt
20	GPIO_INT22	GPIO interrupt
21	GPIO_INT23	GPIO interrupt
22	GPIO_INT24	GPIO interrupt
23	GPIO_INT25	GPIO interrupt
24	GPIO_INT26	GPIO interrupt
25	GPIO_INT27	GPIO interrupt
26	GPIO_INT0	GPIO interrupt
27	GPIO_INT1	GPIO interrupt
28	GPIO_INT2	GPIO interrupt
29	GPIO_INT3	GPIO interrupt
30	GPIO_INT4	GPIO interrupt
31	GPIO_INT5	GPIO interrupt
32	GPIO_INT6	GPIO interrupt
33	GPIO_INT7	GPIO interrupt
34	ARM_NCNTVIRQ3	ARM virtual timer interrupt for core 3
35	ARM_NCNTVIRQ2	ARM virtual timer interrupt for core 2
36	ARM_NCNTVIRQ1	ARM virtual timer interrupt for core 1
37	ARM_NCNTVIRQ0	ARM virtual timer interrupt for core 0
38	CIC_2_OUT48	CIC2 Interrupt Controller output
39	Reserved	Reserved
40	UART_0_URXEVT	UART0 receive event
41	UART_0_UTXEVT	UART0 transmit event
42	CIC_2_OUT22	CIC2 Interrupt Controller output
43	CIC_2_OUT23	CIC2 Interrupt Controller output
44	CIC_2_OUT24	CIC2 Interrupt Controller output
45	CIC_2_OUT25	CIC2 Interrupt Controller output
46	CIC_2_OUT26	CIC2 Interrupt Controller output
47	CIC_2_OUT27	CIC2 Interrupt Controller output
48	CIC_2_OUT28	CIC2 Interrupt Controller output
49	SPI_0_XEVT	SPI0 receive event
50	SPI_0_REVT	SPI0 transmit event
51	Reserved	Reserved
52	ARM_NCNTPNSIRQ3	ARM non secure timer interrupt for Core 3
53	ARM_NCNTPNSIRQ2	ARM non secure timer interrupt for Core 2
54	ARM_NCNTPNSIRQ1	ARM non secure timer interrupt for Core 1
55	ARM_NCNTPNSIRQ0	ARM non secure timer interrupt for Core 0
56	QMSS_QUE_PEND_581	Navigator transmit queue pending event for indicated queue

Table 3-29 EDMA3CC2 Events for AM5K2E04/02 (Part 3 of 3)

Event No.	Event Name	Description
57	QMSS_QUE_PEND_582	Navigator transmit queue pending event for indicated queue
58	QMSS_QUE_PEND_583	Navigator transmit queue pending event for indicated queue
59	QMSS_QUE_PEND_584	Navigator transmit queue pending event for indicated queue
60	QMSS_QUE_PEND_585	Navigator transmit queue pending event for indicated queue
61	QMSS_QUE_PEND_586	Navigator transmit queue pending event for indicated queue
62	QMSS_QUE_PEND_587	Navigator transmit queue pending event for indicated queue
63	QMSS_QUE_PEND_588	Navigator transmit queue pending event for indicated queue
End of Table 3-29		

Table 3-30 EDMA3CC3 Events for AM5K2E04/02 (Part 1 of 2)

Event No.	Event Name	Description
0	USIM_RREQ	USIM read DMA event
1	USIM_WREQ	USIM write DMA event
2	SPI_2_XEVT	SPI2 transmit event
3	SPI_2_REVT	SPI2 receive event
4	I2C_2_REVT	I2C2 receive
5	I2C_2_XEVT	I2C2 transmit
6	UART_1_URXEVT	UART1 receive event
7	UART_1_UTXEVT	UART1 transmit event
8	Reserved	Reserved
9	Reserved	Reserved
10	SPI_1_XEVT	SPI1 transmit event
11	SPI_1_REVT	SPI1 receive event
12	I2C_0_REVT	I2C0 receive
13	I2C_0_XEVT	I2C0 transmit
14	I2C_1_REVT	I2C1 receive
15	I2C_1_XEVT	I2C1 transmit
16	TIMER_16_INTL	Timer interrupt low
17	TIMER_16_INTH	Timer interrupt high
18	TIMER_17_INTL	Timer interrupt low
19	TIMER_17_INTH	Timer interrupt high
20	ARM_TBR_DMA	Debug trace buffer (TBR) DMA event
21	DBGTBR_DMAINT	ARM trace buffer (TBR) DMA event
22	UART_0_URXEVT	UART0 receive event
23	UART_0_UTXEVT	UART0 transmit event
24	GPIO_INT16	GPIO interrupt
25	GPIO_INT17	GPIO interrupt
26	GPIO_INT18	GPIO interrupt
27	GPIO_INT19	GPIO interrupt
28	GPIO_INT20	GPIO interrupt
29	GPIO_INT21	GPIO interrupt
30	GPIO_INT22	GPIO interrupt
31	GPIO_INT23	GPIO interrupt
32	GPIO_INT24	GPIO interrupt

PRODUCT PREVIEW

Table 3-30 EDMA3CC3 Events for AM5K2E04/02 (Part 2 of 2)

Event No.	Event Name	Description
33	GPIO_INT25	GPIO interrupt
34	GPIO_INT26	GPIO interrupt
35	GPIO_INT27	GPIO interrupt
36	GPIO_INT28	GPIO interrupt
37	GPIO_INT29	GPIO interrupt
38	GPIO_INT30	GPIO interrupt
39	GPIO_INT31	GPIO interrupt
40	QMSS_QUE_PEND_589	Navigator transmit queue pending event for indicated queue
41	QMSS_QUE_PEND_590	Navigator transmit queue pending event for indicated queue
42	QMSS_QUE_PEND_591	Navigator transmit queue pending event for indicated queue
43	QMSS_QUE_PEND_592	Navigator transmit queue pending event for indicated queue
44	QMSS_QUE_PEND_593	Navigator transmit queue pending event for indicated queue
45	QMSS_QUE_PEND_594	Navigator transmit queue pending event for indicated queue
46	QMSS_QUE_PEND_595	Navigator transmit queue pending event for indicated queue
47	QMSS_QUE_PEND_596	Navigator transmit queue pending event for indicated queue
48	QMSS_QUE_PEND_597	Navigator transmit queue pending event for indicated queue
49	QMSS_QUE_PEND_598	Navigator transmit queue pending event for indicated queue
50	QMSS_QUE_PEND_599	Navigator transmit queue pending event for indicated queue
51	QMSS_QUE_PEND_600	Navigator transmit queue pending event for indicated queue
52	QMSS_QUE_PEND_601	Navigator transmit queue pending event for indicated queue
53	QMSS_QUE_PEND_602	Navigator transmit queue pending event for indicated queue
54	QMSS_QUE_PEND_603	Navigator transmit queue pending event for indicated queue
55	QMSS_QUE_PEND_604	Navigator transmit queue pending event for indicated queue
56	CIC_2_OUT57	CIC2 Interrupt Controller output
57	CIC_2_OUT50	CIC2 Interrupt Controller output
58	CIC_2_OUT51	CIC2 Interrupt Controller output
59	CIC_2_OUT52	CIC2 Interrupt Controller output
60	CIC_2_OUT53	CIC2 Interrupt Controller output
61	CIC_2_OUT54	CIC2 Interrupt Controller output
62	CIC_2_OUT55	CIC2 Interrupt Controller output
63	CIC_2_OUT56	CIC2 Interrupt Controller output
End of Table 3-30		

Table 3-31 EDMA3CC4 Events for AM5K2E04/02 (Part 1 of 3)

Event No.	Event Name	Description
0	GPIO_INT16	GPIO interrupt
1	GPIO_INT17	GPIO interrupt
2	GPIO_INT18	GPIO interrupt
3	GPIO_INT19	GPIO interrupt
4	GPIO_INT20	GPIO interrupt
5	GPIO_INT21	GPIO interrupt
6	GPIO_INT22	GPIO interrupt
7	GPIO_INT23	GPIO interrupt
8	GPIO_INT24	GPIO interrupt

PRODUCT PREVIEW

Table 3-31 EDMA3CC4 Events for AM5K2E04/02 (Part 2 of 3)

Event No.	Event Name	Description
9	GPIO_INT25	GPIO interrupt
10	GPIO_INT26	GPIO interrupt
11	GPIO_INT27	GPIO interrupt
12	GPIO_INT28	GPIO interrupt
13	GPIO_INT29	GPIO interrupt
14	GPIO_INT30	GPIO interrupt
15	GPIO_INT31	GPIO interrupt
16	Reserved	Reserved
17	SEM_INT8	Semaphore interrupt
18	SEM_INT9	Semaphore interrupt
19	SEM_INT10	Semaphore interrupt
20	SEM_INT11	Semaphore interrupt
21	SEM_INT12	Semaphore interrupt
22	TIMER_12_INTL	Timer interrupt low
23	TIMER_12_INTH	Timer interrupt high
24	TIMER_8_INTL	Timer interrupt low
25	TIMER_8_INTH	Timer interrupt high
26	TIMER_14_INTL	Timer interrupt low
27	TIMER_14_INTH	Timer interrupt high
28	TIMER_15_INTL	Timer interrupt low
29	TIMER_15_INTH	Timer interrupt high
30	DBGTBR_DMAINT	Debug trace buffer (TBR) DMA event
31	ARM_TBR_DMA	ARM trace buffer (TBR) DMA event
32	QMSS_QUE_PEND_658	Navigator transmit queue pending event for indicated queue
33	QMSS_QUE_PEND_659	Navigator transmit queue pending event for indicated queue
34	QMSS_QUE_PEND_660	Navigator transmit queue pending event for indicated queue
35	QMSS_QUE_PEND_661	Navigator transmit queue pending event for indicated queue
36	QMSS_QUE_PEND_662	Navigator transmit queue pending event for indicated queue
37	QMSS_QUE_PEND_663	Navigator transmit queue pending event for indicated queue
38	QMSS_QUE_PEND_664	Navigator transmit queue pending event for indicated queue
39	QMSS_QUE_PEND_665	Navigator transmit queue pending event for indicated queue
40	QMSS_QUE_PEND_605	Navigator transmit queue pending event for indicated queue
41	QMSS_QUE_PEND_606	Navigator transmit queue pending event for indicated queue
42	QMSS_QUE_PEND_607	Navigator transmit queue pending event for indicated queue
43	QMSS_QUE_PEND_608	Navigator transmit queue pending event for indicated queue
44	QMSS_QUE_PEND_609	Navigator transmit queue pending event for indicated queue
45	QMSS_QUE_PEND_610	Navigator transmit queue pending event for indicated queue
46	QMSS_QUE_PEND_611	Navigator transmit queue pending event for indicated queue
47	QMSS_QUE_PEND_612	Navigator transmit queue pending event for indicated queue
48	ARM_NCNTVIRQ3	ARM virtual timer interrupt for Core 3
49	ARM_NCNTVIRQ2	ARM virtual timer interrupt for Core 2
50	ARM_NCNTVIRQ1	ARM virtual timer interrupt for Core 1
51	ARM_NCNTVIRQ0	ARM virtual timer interrupt for Core 0
52	ARM_NCNTPNSIRQ3	ARM non secure timer interrupt for Core 3

Table 3-31 EDMA3CC4 Events for AM5K2E04/02 (Part 3 of 3)

Event No.	Event Name	Description
53	ARM_NCNTPNSIRQ2	ARM non secure timer interrupt for Core 2
54	ARM_NCNTPNSIRQ1	ARM non secure timer interrupt for Core 1
55	ARM_NCNTPNSIRQ0	ARM non secure timer interrupt for Core 0
56	CIC_2_OUT82	CIC2 Interrupt Controller output
57	CIC_2_OUT83	CIC2 Interrupt Controller output
58	CIC_2_OUT84	CIC2 Interrupt Controller output
59	CIC_2_OUT85	CIC2 Interrupt Controller output
60	CIC_2_OUT86	CIC2 Interrupt Controller output
61	CIC_2_OUT87	CIC2 Interrupt Controller output
62	CIC_2_OUT88	CIC2 Interrupt Controller output
63	CIC_2_OUT89	CIC2 Interrupt Controller output
End of Table 3-31		

IMPORTANT NOTICE

Texas Instruments Incorporated and its subsidiaries (TI) reserve the right to make corrections, enhancements, improvements and other changes to its semiconductor products and services per JESD46, latest issue, and to discontinue any product or service per JESD48, latest issue. Buyers should obtain the latest relevant information before placing orders and should verify that such information is current and complete. All semiconductor products (also referred to herein as “components”) are sold subject to TI’s terms and conditions of sale supplied at the time of order acknowledgment.

TI warrants performance of its components to the specifications applicable at the time of sale, in accordance with the warranty in TI’s terms and conditions of sale of semiconductor products. Testing and other quality control techniques are used to the extent TI deems necessary to support this warranty. Except where mandated by applicable law, testing of all parameters of each component is not necessarily performed.

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

TI does not warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right relating to any combination, machine, or process in which TI components or services are used. Information published by TI regarding third-party products or services does not constitute a license 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 significant portions of TI 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. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

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

Buyer acknowledges and agrees that it is solely responsible for compliance with all legal, regulatory and safety-related requirements concerning its products, and any use of TI components in its applications, notwithstanding any applications-related information or support that may be provided by TI. Buyer represents and agrees that it has all the necessary expertise to create and implement safeguards which anticipate dangerous consequences of failures, monitor failures and their consequences, lessen the likelihood of failures that might cause harm and take appropriate remedial actions. Buyer will fully indemnify TI and its representatives against any damages arising out of the use of any TI components in safety-critical applications.

In some cases, TI components may be promoted specifically to facilitate safety-related applications. With such components, TI’s goal is to help enable customers to design and create their own end-product solutions that meet applicable functional safety standards and requirements. Nonetheless, such components are subject to these terms.

No TI components are authorized for use in FDA Class III (or similar life-critical medical equipment) unless authorized officers of the parties have executed a special agreement specifically governing such use.

Only those TI components which TI has specifically designated as military grade or “enhanced plastic” are designed and intended for use in military/aerospace applications or environments. Buyer acknowledges and agrees that any military or aerospace use of TI components which have ***not*** been so designated is solely at the Buyer's risk, and that Buyer is solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI has specifically designated certain components which meet ISO/TS16949 requirements, mainly for automotive use. Components which have not been so designated are neither designed nor intended for automotive use; and TI will not be responsible for any failure of such components to meet such requirements.

Products		**Applications**	
Audio	www.ti.com/audio	Automotive and Transportation	www.ti.com/automotive
Amplifiers	amplifier.ti.com	Communications and Telecom	www.ti.com/communications
Data Converters	dataconverter.ti.com	Computers and Peripherals	www.ti.com/computers
DLP® Products	www.dlp.com	Consumer Electronics	www.ti.com/consumer-apps
DSP	dsp.ti.com	Energy and Lighting	www.ti.com/energy
Clocks and Timers	www.ti.com/clocks	Industrial	www.ti.com/industrial
Interface	interface.ti.com	Medical	www.ti.com/medical
Logic	logic.ti.com	Security	www.ti.com/security
Power Mgmt	power.ti.com	Space, Avionics and Defense	www.ti.com/space-avionics-defense
Microcontrollers	microcontroller.ti.com	Video and Imaging	www.ti.com/video
RFID	www.ti-rfid.com		
OMAP Applications Processors	www.ti.com/omap	**TI E2E Community**	e2e.ti.com
Wireless Connectivity	www.ti.com/wirelessconnectivity		

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