KeyStone Architecture
Network Coprocessor (NETCP)

User Guide

Literature Number: SPRUGZ6
November 2010

Release History

Release	Date	Chapter/Topic	Description/Comments
1.0	November 2010	All	Initial Release

Contents

List of Tables

List of Figures

List of Procedures

Preface

About This Manual

The network coprocessor (NETCP) is a hardware accelerator that processes data packets with a main focus on processing Ethernet packets. NETCP has two gigabit Ethernet (GbE) modules to send and receive packets from an IEEE 802.3 compliant network, a packet accelerator (PA) to perform packet classification operations such as header matching, and packet modification operations such as checksum generation, A and a security accelerator (SA) to encrypt and decrypt data packets.

Notational Conventions

This document uses the following conventions:

- Commands and keywords are in **boldface** font.
- Arguments for which you supply values are in *italic* font.
- Terminal sessions and information the system displays are in `screen font`.
- Information you must enter is in **`boldface screen font`**.
- Elements in square brackets ([ ]) are optional.

Notes use the following conventions:

Note—Means reader take note. Notes contain helpful suggestions or references to material not covered in the publication.

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

CAUTION—Indicates the possibility of service interruption if precautions are not taken.

WARNING—Indicates the possibility of damage to equipment if precautions are not taken.

Related Documentation from Texas Instruments

Gigabit Ethernet (GbE) Switch Subsystem for KeyStone Devices User Guide	SPRUGV9
Interrupt Controller (INTC) for KeyStone Devices User Guide	SPRUGW4
Multicore Navigator for KeyStone Devices User Guide	SPRUGR9
Packet Accelerator (PA) for KeyStone Devices User Guide	SPRUGS4
Security Accelerator (SA) for KeyStone Devices User Guide	SPRUGY6

Trademarks

Chapter 1

Introduction

1.1 Purpose of the Peripheral

The network coprocessor (NETCP) is a hardware accelerator that processes data packets with a main focus on processing Ethernet packets. NETCP has two gigabit Ethernet (GbE) modules to send and receive packets from an IEEE 802.3 compliant network. The NETCP also includes a packet accelerator (PA) to perform packet classification operations such as header matching, and packet modification operations such as checksum generation. The NETCP also provides a security accelerator (SA) to encrypt and decrypt data packets. The NETCP can receive packets from the Ethernet modules, or packets can be delivered to the NETCP through packet DMA from the DSP or another supported peripheral (such as SRIO).

1.2 Terminology Used in This Document

The following acronyms and abbreviations appear in this user guide.

Term	Definition
GbE	Gigabit Ethernet
MAC	Media Access Controller
MDIO	Management Data Input/Output
NETCP	Network Coprocessor
PA	Packet Accelerator
PKTDMA	Packet DMA
QMSS	Queue Manager subsystem
SA	Security Accelerator
SerDes	Serializer/Deserializer
SGMII	Serial Gigabit Media Independent Interface
SRIO	Serial RapidIO

1.3 Features

The NETCP has the following features:

- Packet DMA controller for interfacing to the Queue Manager subsystem
  - 9 packet DMA transmit channels
  - 24 packet DMA receive channels
  - 32 receive flows
- Packet accelerator for packet header processing operations
  - Layer 2 processing engine
    - MAC processing
  - (2) Layer 3 processing engines
    - IPv4, IPv6, and custom layer 3
  - Layer 4 processing engine
    - UDP, TCP, custom layer 4
  - (2) Modify/Multiroute processing engines

- Security accelerator for encryption and decryption operations
  - IPSEC protocol stack
  - SRTP protocol stack
  - 3GPP protocol stack, Wireless Air cipher standard
  - True random number generator
  - Public key accelerator
- Gigabit Ethernet Switch subsystem for interfacing to an 802.3 compliant Ethernet network
  - (2) one-gigabit Serial Gigabit Media Independent Interface (SGMII) modules
  - Three-port gigabit Ethernet switch
  - Time Synchronization for compliance with IEEE 1588

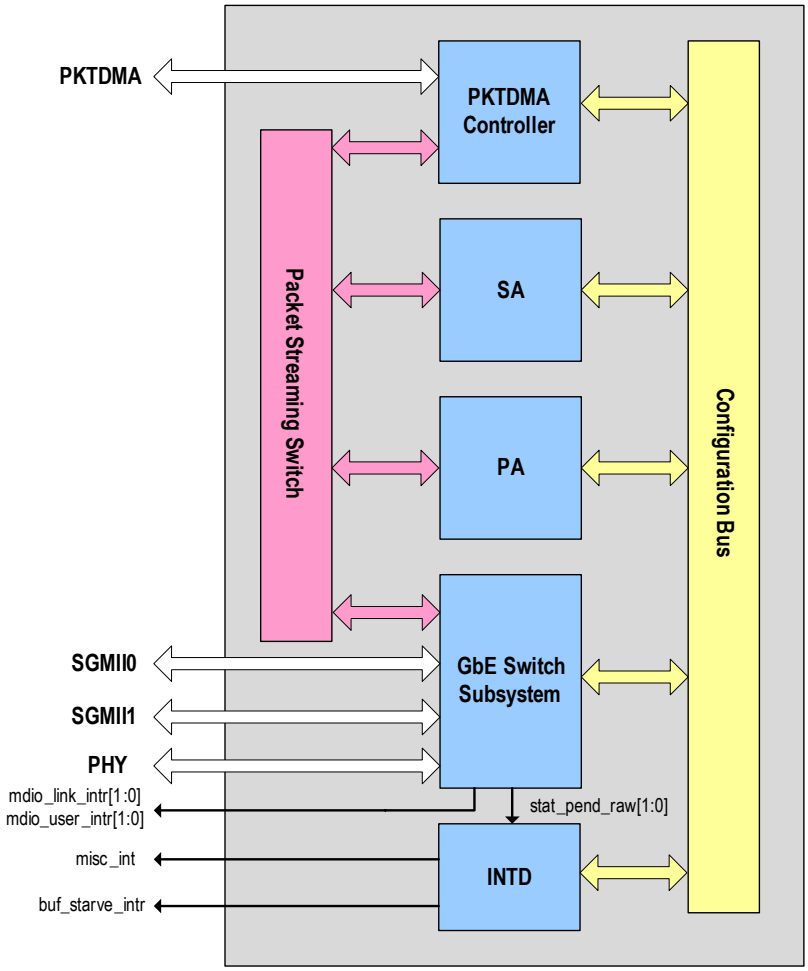
1.4 Functional Block Diagram

Figure 1-1 shows the network coprocessor (NETCP) functional block diagram. The NETCP has four major modules that are connected through the packet streaming switch.

The four major modules are:

- Packet DMA (PKTDMA) controller
- Packet Accelerator (PA)
- Security Accelerator (SA)
- Gigabit Ethernet (GbE) Switch subsystem

Figure 1-1 Network Coprocessor Functional Block Diagram



1.5 Industry Standard(s) Compliance Statement

The modules in the NETCP support a variety of standards, which are covered in detail in the respective module user guides. For more information about the standards supported by each module, see the following:

- Packet Accelerator User Guide
- Security Accelerator User Guide
- Gigabit Ethernet Switch Subsystem User Guide
- Multicore Navigator User Guide (*for Packet DMA*)

Chapter 2

Architecture

2.1 Clock Control

The network coprocessor (NETCP) has three primary clock domains:

1. Packet Accelerator (PA)
2. Security Accelerator (SA)
3. Gigabit Ethernet (GbE) Switch subsystem

Each of these three clock domains share a common source clock, which is expected to operate at 350 MHz. Before using the PA, SA, or GbE switch modules, the respective clock domain must be enabled. In some devices, this clock may be generated from a PLL. For more information about generating and enabling these clocks, as well as if a PLL is used, see the device-specific data manual.

The NETCP also has secondary clocks, which are used exclusively by the GbE switch subsystem for time synchronization, MDIO, and SGMII SerDes interfaces. For more information about selecting and configuring those clocks, see the *GbE Switch Subsystem User Guide* in "Related Documentation from Texas Instruments" on page ø-x.

2.2 Memory Map

Table 2-1 shows the memory map of the network coprocessor (NETCP).

Table 2-1 Network Coprocessor Memory Map

Region	Address Offset[1]
Packet Accelerator	00000h
Packet Streaming Switch	00600h
Reserved	00610h
Packet Accelerator (continued)	01000h
Packet DMA	04000h
Packet Accelerator Statistics	06000h
Reserved	06100h
Distributed Interrupt Controller	07000h
Reserved	07400h
Gigabit Ethernet Switch Subsystem	90000h
Security Accelerator	C0000h

1. These register address offsets are relative to the base of the NETCP module. See the device- specific data manual to determine the base of the NETCP module.

2.3 Protocol Description(s)

See the module-specific user guides for the supported protocols in "Related Documentation from Texas Instruments" on page ø-x.

2.4 Packet Streaming Switch Architecture

This section gives an overview of the packet streaming switch provided in NETCP. The purpose of the packet streaming switch is to provide a means of transferring data between modules within NETCP. Table 2-2 shows all connections between the modules provided by the packet streaming switch.

Table 2-2 Packet Streaming Switch Module Connections

	Transmit Ports										
Receive Ports	**PDSP0 Out**	**PDSP1 Out**	**PDSP2 Out**	**PDSP3 Out**	**PDSP4 Out**	**PDSP5 Out**	**SA0 Out**	**SA1 Out**	**Ethernet Switch Tx A[1]**	**Ethernet Switch Tx B[2]**	**PKTDMA Tx**
PDSP0 In		1	1	1	1	1			1	1	1
PDSP1 In	1		1	1	1	1			1	1	1
PDSP2 In	1	1		1	1	1			1	1	1
PDSP3 In	1	1	1		1	1			1	1	1
PDSP4 In	1	1	1	1		1			1	1	1
PDSP5 In	1	1	1	1	1				1	1	1
SA0 In	1	1	1	1	1	1					1
SA1 In	1	1	1	1	1	1					1
Ethernet Switch Rx	1	1	1	1	1	1					1
PKTDMA DMA Rx	1	1	1	1	1	1	1	1	1	1	
End of Table 2-0											

1. The Ethernet Switch TX A port can only map to one receive port. Which receive port is used is configurable through the TXSTA field in the packet streaming switch CPSW configuration register.
2. The Ethernet Switch TX B port can only map to one receive port. Which receive port is used is configurable through the TXSTB field in the packet streaming switch CPSW configuration register.

In general, the packet accelerator (PA), security accelerator (SA), and gigabit Ethernet (GbE) modules should communicate only through the queue manager subsystem (QMSS). These modules should not communicate directly because the limited buffering in the system can degrade performance.

Performance degradation can occur when one module has finished processing a packet and needs to forward the packet to a second module for further processing. If the second module is still processing a packet, the first module must hold the packet until the second module has completed processing its packet. While the first module is holding the packet, it cannot process any other packets. The solution is to direct the first module to send its packet to the transmit queue of another module. This way, one module can send a packet to another module through the QMSS with no interaction from the host, while allowing modules to process packets independent of the status of other modules. Please note that it is okay for submodules within a module to communicate with each other without using the queue manager. For example the PDSP submodules in the PA can communicate with each other directly without using the queue manager, but if a PA PDSP wants to route a packet to the SA for processing, it must use the queue manager.

The following example describes an instance of potential performance degradation. A SRTP packet with MAC, IPv4, and UDP headers arrives from the GbE switch. The processing steps are as follows:

1. The packet arrives from the GbE switch and is sent to the PDSP0 to match the MAC header
2. The packet is then routed to PDSP1 to match the IPv4 header

3. The packet is routed to PDSP3 to match the UDP header
4. The packet is routed to the SA to decrypt the packet

If PDSP3 tries to send the packet directly to the SA, but the SA is currently decrypting a packet—which may take longer than it takes one of the PA PDSPs to match a packet header—PDSP3 must hold the packet until the SA has finished processing the other packet. If PDSP3 holds its packet, PDSP1 and PDSP0 must also hold their packets, which causes the GbE switch to have to hold its packets, causing the entire pipeline to stall. Furthermore, if the switch runs out of buffering, it will start dropping packets.

If—after matching the UDP header—PDSP3 sends its packet to the QMSS instead of holding the packet and waiting for the SA to become available, PDSP3 can start processing another packet immediately and the pipeline does not stall. Furthermore, if PDSP3 sends the packet to the transmit queue for the SA, as soon as the SA finishes processing its packet the packet DMA transfers the next packet in the queue to the SA automatically! This method results in no stalls or downtime in the NETCP, giving better overall performance. Furthermore, all interactions between modules were done using the QMSS and packet DMA without interaction from the host, just by directing PDSP3 to use the SA transmit queue as its receive queue.

The exceptions to this rule of not transferring packets between modules are as follows:

1. Transferring packets between PDSPs in the PA—In this case, packets typically form a "pipeline" where each packet will generally choose the same path through the pipe, and the processing time is roughly equal. This leads to relatively few stalls.
2. Transferring packets between the GbE switch TXSTA and TXSTB interface and the PA—If the packets are entering the NETCP through the GbE switch, the GbE switch can send packets directly to the first PDSP that will be processing the packets without using the QMSS.

2.5 Packet DMA Architecture

The packet DMA controller in NETCP is responsible for transferring data between NETCP and the Host. Data received from the Host can be transmitted via the packet streaming switch to the receive ports of devices within the NETCP as listed in Table 2-2. Data can be received by the packet DMA controller via the packet streaming switch from the transmit ports of devices as listed in Table 2-2. For more information about packet DMA, see the *Multicore Navigator User Guide in* "Related Documentation from Texas Instruments" on page ø-x.

2.6 Distributed Interrupt Controller Architecture

The distributed interrupt controller in NETCP has the ability to aggregate interrupts from several sources and combine these interrupts into one interrupt to the host.

2.7 Reset Considerations

The gigabit Ethernet (GbE) switch subsystem supports reset isolation. For more information regarding this functionality, see the *GbE Switch Subsystem User Guide* in "Related Documentation from Texas Instruments" on page ø-x.

2.8 Initialization

This section describes how to initialize the network coprocessor (NETCP).

The NETCP relies on the queue manager subsystem (QMSS) and the packet DMA to communicate with the host, which requires that these two systems be set up before setting up the network coprocessor. After setting up the QMSS and the packet DMA, the user can configure the NETCP. The user should take care to program the packet streaming switch CPSW configuration register so that it directs packets from the gigabit Ethernet (GbE) switch subsystem to the desired module. If using the packet accelerator (PA) and security accelerator (SA) together, the PA must be initialized before initializing the SA. Otherwise, the PA, SA, and GbE switch subsystem can be initialized in any order.

Procedure 2-1 Network Coprocessor Initialization Procedure

Step	-	Action
1		Turn on the NETCP power domain.
2		Ungate the clocks for all modules used.
3		Configure the queue manager.
4		Configure the packet DMA.
	4a	Configure the linking RAM.
	4b	Initialize descriptors.
	4c	Configure receive flows.
	4d	Enable transmit channels.
	4e	Enable receive channels.
5		Configure the packet streaming switch.
6		Configure the GbE switch subsystem.
7		Configure the PA.
8		Configure the SA.

End of Procedure 2-1

2.9 Interrupt Support

2.9.1 Interrupt Events and Requests

The NETCP supports interrupts from the gigabit Ethernet (GbE) switch subsystem and the packet accelerator (PA). For more information about the interrupts generated by the GbE switch subsystem, see the *GbE Switch Subsystem for KeyStone Devices User Guide* in "Related Documentation from Texas Instruments" on page ø-x. For more information about the interrupts generated by the PA, see the P*acket Accelerator (PA) for KeyStone Devices User Guide* in "Related Documentation from Texas Instruments" on page ø-x.

2.9.2 Interrupt Multiplexing

Interrupt multiplexing is supported on the NETCP through the distributed interrupt controller. For more information, see the distributed interrupt controller section.

2.10 Power Management

The network coprocessor (NETCP) has power management capabilities for powering down the NETCP power domain. The power domain is powered on or powered off in its entirety; it does not support the ability to power down the individual modules in the NETCP. By default, the NETCP power domain is powered off. To use the NETCP, the power domain must be powered on.

In addition to allowing the disabling the NETCP power domain, NETCP also allows the clocks to the unused modules to be gated. The packet accelerator (PA), security accelerator (SA), and gigabit Ethernet (GbE) switch subsystem can each be clock-gated individually. This allows any unused modules in NETCP to be clock gated without effecting the other modules in the NETCP.

For more information, see the device-specific data manual.

Note—The packet DMA controller and the packet streaming switch are included in the packet accelerator clock domain. This means that the packet accelerator clock domain must be enabled when using the security accelerator or the gigabit Ethernet switch subsystem.

Chapter 3

Registers

This chapter describes the registers in the network coprocessor (NETCP). For clarity, the registers are presented for each module. Each register has a bit-field description and a memory offset address. The offset address values are relative to the base address of the specific module. The base address of the module can be found by referencing the NETCP memory map. See the device-specific data manual for the base address NETCP.

3.1 Packet Streaming Switch Register Region

This section describes the packet streaming interface switch register region.

Table 3-1 Packet Streaming Switch Register Region

Address Offset	Register	Section
000h	Version and Identification Register	Section 3-2
004h	CPSW Configuration Register	Section 3-3

3.1.1 Version and Identification Register (REVISION_REG)

The version and identification register is shown in Table 3-2.

Table 3-2 Version and Identification Register

Bit	Name	Type	Reset	Description
31-30	REV_SCHEME	R	0	Scheme Revision.
27-16	REV_MODULE	R	0	Module Revision.
15-11	REV_RTL	R	0	RTL Revision.
10-8	REV_MAJOR	R	0	Major Revision.
7-6	REV_CUSTOM	R	0	Custom Revision.
5-0	REV_MINOR	R	0	Minor Revision.

3.1.2 CPSW Configuration Register (CPSW_CFG_REG)

The CPSW Configuration Register is shown in Table 3-3.

Table 3-3 CPSW Configuration Register

Bit	Name	Type	Reset	Description
31-11	Reserved	R	0	Reserved.
10-8	TXSTB	RW	6	This field maps the CPSW TXSTB port to a packet DMA channel. Register default value is 6. 0 = PDSP0 1 = PDSP1 2 = PDSP2 3 = PDSP3 4 = PDSP4 5 = PDSP5 6 = Packet DMA channel 23
7-3	Reserved	R	0	Reserved.
2-0	TXSTA	RW	6	This field maps the CPSW TXSTA port to a packet DMA channel. Register default value is 6. 0 = PDSP0 1 = PDSP1 2 = PDSP2 3 = PDSP3 4 = PDSP4 5 = PDSP5 6 = Packet DMA channel 22

3.2 Distributed Interrupt Controller Register Region

This section describes the distributed interrupt controller (INTD) registers available in the network coprocessor (NETCP). The register address offsets listed in Table 3-4 are relative to the base of INTD memory region. To determine the base address of INTD memory region relative to the NETCP memory map, see Table 2-1 "Network Coprocessor Memory Map" on page 2-2.

Table 3-4 Distributed Interrupt Controller Register Region Memory Map

Address Offset[1]	Register	Section
000h	Revision Register	Section 3.2.1
004h	Control Register	Section 3.2.2
008h	Reserved	Reserved
010h	EOI Register	Section 3.2.3
014h	EOI Interrupt Vector Register	Section 3.2.4
018h - 1FCh	Reserved	Reserved
200h - 27Ch	Status Register 0	Section 3.2.5
204h	Status Register 1	Section 3.2.6
208h	Status Register 2	Section 3.2.7
20Ch – 27Ch	Reserved	Reserved
280h – 2FCh	Status Clear Register 0	Section 3.2.8
284h	Status Clear Register 1	Section 3.2.9
288h	Status Clear Register 2	Section 3.2.10
28Ch-2FCh	Reserved	Reserved
300h	Interrupt Counter Register 0	Section 3.2.11
304h	Interrupt Counter Register 1	Section 3.2.11
...	...	...
300h + (N*4)	Interrupt Counter Register N	Section 3.2.11
...	...	...
380h	Interrupt Counter Register for host_cnt_int32_cdma_starve	Section 3.2.12
380h – 47Ch	Reserved	Reserved
480h	Interrupt Vector Registers	Section 3.2.13
484h – 4FCh	Reserved	Reserved
End of Table 3-4		

1. The address offsets listed in this table are relative to the base of the IND memory region.

3.2.1 Revision Register

The Revision Register contains the identification and revision information.

Table 3-5 Revision Register

Bits	Field	Type	Reset	Description
31-30	Reserved	R	0h	Always read as 0. Writes have no affect.
29-16	MODID	R	X	Module ID field.
15-11	REVRTL	R	Any	RTL revision. Will vary depending on release.
10-8	REVMAJ	R	1h	Major revision.
7-0	REVMIN	R	0h	Minor revision.
End of Table 3-5				

3.2.2 Control Register

The Control Register allows software control of the module.

Table 3-6 Control Register

Bits	Field	Type	Reset	Description
31-0	Reserved	R	0h	Always read as 0. Writes have no affect.
End of Table 3-6				

3.2.3 EOI Register

The EOI Register allows the software to perform an end of interrupt handshake to interrupts with an IP that does not have an EOI register. The EOI handshake allows system interrupts to be re-evaluated for the associate logic with the EOI vector value (to allow multiple logic paths to re-evaluate individually without separate EOI registers). The EOI occurs the cycle after the register is written.

Table 3-7 EOI Register

Bits	Field	Type	Reset	Description
31-8	Reserved	R	0h	Always read as 0. Writes have no effect.
7-0	EOI_VECTOR	RW	0h	EOI Vector value. This should be written with the value associate to the interrupt and host combination given in the INTD configuration.
End of Table 3-7				

3.2.4 EOI Interrupt Vector Register

The EOI Interrupt Vector Register captures the active and prioritized interrupts so that software may quickly read the values rather than check every bit in the IP Interrupt Source Register.

Table 3-8 EOI Interrupt Vector Register

Bits	Field	Type	Reset	Description
31-0	INTR_VECTOR	R	0h	EOI Interrupt Vector value. This has the latest prioritized interrupt values.
End of Table 3-8				

3.2.5 Status Register 0

The Status Register indicates which IP interrupts are active for the host.

Table 3-9 Status Register 0 (Part 1 of 3)

Bit	Name	Type	Reset	Description
31	STATUS_HOST_IN_INTR31	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
30	STATUS_HOST_IN_INTR30	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
29	STATUS_HOST_IN_INTR29	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
28	STATUS_HOST_IN_INTR28	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
27	STATUS_HOST_IN_INTR27	RW	0	Interrupt Status. 0 = Not pending 1 = Pending

Table 3-9 Status Register 0 (Part 2 of 3)

Bit	Name	Type	Reset	Description
26	STATUS_HOST_IN_INTR26	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
25	STATUS_HOST_IN_INTR25	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
24	STATUS_HOST_IN_INTR24	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
23	STATUS_HOST_IN_INTR23	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
22	STATUS_HOST_IN_INTR22	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
21	STATUS_HOST_IN_INTR21	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
20	STATUS_HOST_IN_INTR20	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
19	STATUS_HOST_IN_INTR19	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
18	STATUS_HOST_IN_INTR18	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
17	STATUS_HOST_IN_INTR17	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
16	STATUS_HOST_IN_INTR16	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
15	STATUS_HOST_IN_INTR15	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
14	STATUS_HOST_IN_INTR14	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
13	STATUS_HOST_IN_INTR13	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
12	STATUS_HOST_IN_INTR12	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
11	STATUS_HOST_IN_INTR11	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
10	STATUS_HOST_IN_INTR10	RW	0	Interrupt Status. 0 = Not pending 1 = Pending

Table 3-9 Status Register 0 (Part 3 of 3)

Bit	Name	Type	Reset	Description
9	STATUS_HOST_IN_INTR09	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
8	STATUS_HOST_IN_INTR08	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
7	STATUS_HOST_IN_INTR07	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
6	STATUS_HOST_IN_INTR06	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
5	STATUS_HOST_IN_INTR05	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
4	STATUS_HOST_IN_INTR04	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
3	STATUS_HOST_IN_INTR03	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
2	STATUS_HOST_IN_INTR02	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
1	STATUS_HOST_IN_INTR01	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
0	STATUS_HOST_IN_INTR00	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
End of Table 3-9				

3.2.6 Status Register 1

The Status Register 1 indicates which IP interrupts are active for the host.

Table 3-10 Status Register 1s (Part 1 of 3)

Bit	Name	Type	Reset	Description
31	STATUS_HOST_INT31_LINTR31	R	0	Interrupt Status. 0 = Not pending 1 = Pending
30	STATUS_HOST_INT30_LINTR30	R	0	Interrupt Status. 0 = Not pending 1 = Pending
29	STATUS_HOST_INT29_LINTR29	R	0	Interrupt Status. 0 = Not pending 1 = Pending
28	STATUS_HOST_INT28_LINTR28	R	0	Interrupt Status. 0 = Not pending 1 = Pending
27	STATUS_HOST_INT27_LINTR27	R	0	Interrupt Status. 0 = Not pending 1 = Pending

Table 3-10 Status Register 1s (Part 2 of 3)

Bit	Name	Type	Reset	Description
26	STATUS_HOST_INT26_LINTR26	R	0	Interrupt Status. 0 = Not pending 1 = Pending
25	STATUS_HOST_INT25_LINTR25	R	0	Interrupt Status. 0 = Not pending 1 = Pending
24	STATUS_HOST_INT24_LINTR24	R	0	Interrupt Status. 0 = Not pending 1 = Pending
23	STATUS_HOST_INT23_LINTR23	R	0	Interrupt Status. 0 = Not pending 1 = Pending
22	STATUS_HOST_INT22_LINTR22	R	0	Interrupt Status. 0 = Not pending 1 = Pending
21	STATUS_HOST_INT21_LINTR21	R	0	Interrupt Status. 0 = Not pending 1 = Pending
20	STATUS_HOST_INT19_LINTR19	R	0	Interrupt Status. 0 = Not pending 1 = Pending
19	STATUS_HOST_INT19_LINTR19	R	0	Interrupt Status. 0 = Not pending 1 = Pending
18	STATUS_HOST_INT18_LINTR18	R	0	Interrupt Status. 0 = Not pending 1 = Pending
17	STATUS_HOST_INT17_LINTR17	R	0	Interrupt Status. 0 = Not pending 1 = Pending
16	STATUS_HOST_INT16_LINTR16	R	0	Interrupt Status. 0 = Not pending 1 = Pending
15	STATUS_HOST_INT15_LINTR15	R	0	Interrupt Status. 0 = Not pending 1 = Pending
14	STATUS_HOST_INT14_LINTR14	R	0	Interrupt Status. 0 = Not pending 1 = Pending
13	STATUS_HOST_INT13_LINTR13	R	0	Interrupt Status. 0 = Not pending 1 = Pending
12	STATUS_HOST_INT12_LINTR12	R	0	Interrupt Status. 0 = Not pending 1 = Pending
11	STATUS_HOST_INT11_LINTR11	R	0	Interrupt Status. 0 = Not pending 1 = Pending
10	STATUS_HOST_INT10_LINTR10	R	0	Interrupt Status. 0 = Not pending 1 = Pending

Table 3-10 Status Register 1s (Part 3 of 3)

Bit	Name	Type	Reset	Description
9	STATUS_HOST_INT09_LINTR09	R	0	Interrupt Status. 0 = Not pending 1 = Pending
8	STATUS_HOST_INT08_LINTR08	R	0	Interrupt Status. 0 = Not pending 1 = Pending
7	STATUS_HOST_INT07_LINTR07	R	0	Interrupt Status. 0 = Not pending 1 = Pending
6	STATUS_HOST_INT06_LINTR06	R	0	Interrupt Status. 0 = Not pending 1 = Pending
5	STATUS_HOST_INT05_LINTR05	R	0	Interrupt Status. 0 = Not pending 1 = Pending
4	STATUS_HOST_INT04_LINTR04	R	0	Interrupt Status. 0 = Not pending 1 = Pending
3	STATUS_HOST_INT03_LINTR03	R	0	Interrupt Status. 0 = Not pending 1 = Pending
2	STATUS_HOST_INT02_LINTR02	R	0	Interrupt Status. 0 = Not pending 1 = Pending
1	STATUS_HOST_INT01_LINTR01	R	0	Interrupt Status. 0 = Not pending 1 = Pending
0	STATUS_HOST_INT00_LINTR00	R	0	Interrupt Status. 0 = Not pending 1 = Pending
End of Table 3-10				

3.2.7 Status Register 2

The Status Register 2 indicates which IP interrupts are active for the host.

Table 3-11 Status Register 2 (Part 1 of 2)

Bit	Name	Type	Reset	Description
31-5	Reserved	R	0	Reserved.
4	STATUS_HOST_INT36_STAT_OVER	R	0	Interrupt Status. 0 = Not pending 1 = Pending
3	STATUS_HOST_INT35_EVNT_PEND_SYNCD	R	0	Interrupt Status. 0 = Not pending 1 = Pending
2	STATUS_HOST_INT34_CPSW_STAT1_SYNCD	R	0	Interrupt Status. 0 = Not pending 1 = Pending

Table 3-11 Status Register 2 (Part 2 of 2)

Bit	Name	Type	Reset	Description
1	STATUS_HOST_INT33_CPSW_STAT0_SYNCD	R	0	Interrupt Status. 0 = Not pending 1 = Pending
0	STATUS_HOST_INT32_CDMA_STARVE	RW	0	Interrupt Status. 0 = Not pending 1 = Pending
End of Table 3-11				

3.2.8 Status Clear Register 0

The Status Clear Register 0 allows software to clear the pending status.

Table 3-12 Status Clear Register 0 (Part 1 of 3)

Bit	Name	Type	Reset	Description
31	STATUS_HOST_IN_INTR31_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
30	STATUS_HOST_IN_INTR30_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
29	STATUS_HOST_IN_INTR29_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
28	STATUS_HOST_IN_INTR28_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
27	STATUS_HOST_IN_INTR27_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
26	STATUS_HOST_IN_INTR26_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
25	STATUS_HOST_IN_INTR25_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
24	STATUS_HOST_IN_INTR24_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
23	STATUS_HOST_IN_INTR23_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.

Table 3-12 Status Clear Register 0 (Part 2 of 3)

Bit	Name	Type	Reset	Description
22	STATUS_HOST_IN_INTR22_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
21	STATUS_HOST_IN_INTR21_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
20	STATUS_HOST_IN_INTR20_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
19	STATUS_HOST_IN_INTR19_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
18	STATUS_HOST_IN_INTR18_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
17	STATUS_HOST_IN_INTR17_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
16	STATUS_HOST_IN_INTR16_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
15	STATUS_HOST_IN_INTR15_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
14	STATUS_HOST_IN_INTR14_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
13	STATUS_HOST_IN_INTR13_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
12	STATUS_HOST_IN_INTR12_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
11	STATUS_HOST_IN_INTR11_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.

Table 3-12 Status Clear Register 0 (Part 3 of 3)

Bit	Name	Type	Reset	Description
10	STATUS_HOST_IN_INTR10_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
9	STATUS_HOST_IN_INTR09_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
8	STATUS_HOST_IN_INTR08_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
7	STATUS_HOST_IN_INTR07_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
6	STATUS_HOST_IN_INTR06_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
5	STATUS_HOST_IN_INTR05_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
4	STATUS_HOST_IN_INTR04_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
3	STATUS_HOST_IN_INTR03_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
2	STATUS_HOST_IN_INTR02CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
1	STATUS_HOST_IN_INTR01_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
0	STATUS_HOST_IN_INTR00_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.
End of Table 3-12				

3.2.9 Status Clear Register 1

The Status Clear Register 1 is shown in Section 3-13.

Table 3-13 Status Clear Register 1

Bit	Name	Type	Reset	Description
31-0	Reserved	R	0	Reserved.

3.2.10 Status Clear Register 2

The Status Clear Register 2 allows software to clear the pending status.

Table 3-14 Status Clear Register 2

Bit	Name	Type	Reset	Description
31-1	Reserved	R	0	Reserved.
0	STATUS_HOST_INT32_CDMA_STARVE_CLR	RW	0	Interrupt Status. 0 = Not pending 1 = Pending Write 1 to clear.

3.2.11 Interrupt Counter Register N for HOST_CNT_IN_INTRN

The Interrupt Counter Registers indicate the number of pending IP interrupts for the host. The count increments automatically when an input interrupt is detected. Writes subtracts the written data from the count or clear when a 0 is written. The configuration for the Interrupt Counter registers are shown in table Table 3-15.

Table 3-15 Interrupt Counter Register N

Bit	Name	Type	Reset	Description
31-8	RESERVED	R	0h	Always read as 0. Writes have no effect.
7-0	COUNT	RW	0h	Interrupt Count. Read returns the counter value. Writing a non-zero value subtracts that value from the count. Writing a 0 clears the count.

3.2.12 Interrupt Counter Register for HOST_CNT_INT32_CDMA_STARVE

The Interrupt Counter Register indicates the number of pending IP interrupts for the host. The count increments automatically when an input interrupt is detected. Writes subtracts the written data from the count or clear when a 0 is written.

Table 3-16 Interrupt Counter Register for HOST_CNT_INT32_CDMA_STARVE

Bits	Field	Type	Reset	Description
31-8	RESERVED	R	0h	Always read as 0. Writes have no effect.
7-0	COUNT	RW	0h	Interrupt Count. Read returns the counter value. Writing a non-zero value subtracts that value from the count. Writing a 0 clears the count.
End of Table 3-16				

3.2.13 Interrupt Vector Registers

The Interrupt Vector Registers captures the interrupt vector of active and prioritized interrupts for each host based in that hosts' enabled interrupts.

Table 3-17 Interrupt Vector Registers

Bits	Field	Type	Reset	Description
31-0	INTR_VECTOR	R	0h	Interrupt Vector. Value is configuration dependent.
End of Table 3-17				

Index

IMPORTANT NOTICE

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

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

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

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

Reproduction of 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. Reproduction of this information with alteration is an unfair and deceptive business practice. TI is not responsible or liable for such altered documentation. Information of third parties may be subject to additional restrictions.

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

TI products are not authorized for use in safety-critical applications (such as life support) where a failure of the TI product would reasonably be expected to cause severe personal injury or death, unless officers of the parties have executed an agreement specifically governing such use. Buyers represent that they have all necessary expertise in the safety and regulatory ramifications of their applications, and acknowledge and agree that they are solely responsible for all legal, regulatory and safety-related requirements concerning their products and any use of TI products in such safety-critical applications, notwithstanding any applications-related information or support that may be provided by TI. Further, Buyers must fully indemnify TI and its representatives against any damages arising out of the use of TI products in such safety-critical applications.

TI products are neither designed nor intended for use in military/aerospace applications or environments unless the TI products are specifically designated by TI as military-grade or "enhanced plastic." Only products designated by TI as military-grade meet military specifications. Buyers acknowledge and agree that any such use of TI products which TI has not designated as military-grade is solely at the Buyer's risk, and that they are solely responsible for compliance with all legal and regulatory requirements in connection with such use.

TI products are neither designed nor intended for use in automotive applications or environments unless the specific TI products are designated by TI as compliant with ISO/TS 16949 requirements. Buyers acknowledge and agree that, if they use any non-designated products in automotive applications, TI will not be responsible for any failure to meet such requirements.

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

Products		**Applications**	
Amplifiers	amplifier.ti.com	Audio	www.ti.com/audio
Data Converters	dataconverter.ti.com	Automotive	www.ti.com/automotive
DLP® Products	www.dlp.com	Communications and Telecom	www.ti.com/communications
DSP	dsp.ti.com	Computers and Peripherals	www.ti.com/computers
Clocks and Timers	www.ti.com/clocks	Consumer Electronics	www.ti.com/consumer-apps
Interface	interface.ti.com	Energy	www.ti.com/energy
Logic	logic.ti.com	Industrial	www.ti.com/industrial
Power Mgmt	power.ti.com	Medical	www.ti.com/medical
Microcontrollers	microcontroller.ti.com	Security	www.ti.com/security
RFID	www.ti-rfid.com	Space, Avionics & Defense	www.ti.com/space-avionics-defense
RF/IF and ZigBee® Solutions	www.ti.com/lprf	Video and Imaging	www.ti.com/video
		Wireless	www.ti.com/wireless-apps

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