# GLINT *Gamma2*®

## REFERENCE GUIDE

**DRAFT ONLY**

**PROPRIETARY AND CONFIDENTIAL INFORMATION**

3D*labs*®

# GLINT *Gamma2*®

# *Reference Guide*™

**PROPRIETARY AND CONFIDENTIAL INFORMATION**

**Issue 1**

# Proprietary Notice

The material in this document is the intellectual property of **3D***labs*. It is provided solely for information. You may not reproduce this document in whole or in part by any means. While every care has been taken in the preparation of this document, **3D***labs* accepts no liability for any consequences of its use. Our products are under continual improvement and we reserve the right to change their specification without notice. **3D***labs* may not produce printed versions of each issue of this document. The latest version will be available from the **3D***labs* web site.

**3D***labs* products and technology are protected by a number of worldwide patents. Unlicensed use of any information contained herein may infringe one or more of these patents and may violate the appropriate patent laws and conventions.

**3D***labs* is the worldwide trading name of **3D***labs* Inc. Ltd.

**3D***labs*, GLINT Gamma and GLINT are registered trademarks of **3D***labs* Inc. Ltd.

Microsoft, Windows and Direct3D are either registered trademarks or trademarks of Microsoft Corp. in the United States and/or other countries. OpenGL is a registered trademark of Silicon Graphics, Inc. All other trademarks are acknowledged and recognized.



Email: info@3dlabs.com
Web: http://www.3dlabs.com

**3D***labs* Ltd.
Meadlake Place
Thorpe Lea Road, Egham
Surrey, TW20 8HE
United Kingdom
Tel: +44 (0) 1784 470555
Fax: +44 (0) 1784 470699

**3D***labs* K.K
Shiroyama JT Mori Bldg 16F
40301 Toranomon
Minato-ku, Tokyo, 105, Japan
Tel: +81-3-5403-4653
Fax: +91-3-5403-4646

**3D***labs* Inc.
480 Potrero Avenue
Sunnyvale, CA 94086,
United States
Tel: (408) 530-4700
Fax: (408) 530-4701

# Change History

| Document | Issue | Date | Change |
|---|---|---|---|
| 149.6.0 | 1 | 15 July 99 | First Issue. |

# Errata and Alerts

Refer to Release 4 or later releases of the common Gamma 1 / Gamma 2 Errata and Alerts document.

# Contents

# 1 Introduction

This document has been written as the reference for hardware and system designers who wish to develop hardware or software using the GLINT® Gamma geometry processor. A familiarity with the functionality of the GLINT rendering devices is assumed in this document.

## 1.1 What is the GLINT Gamma2?

GLINT Gamma2 is a lighting and geometry processor, designed to break the 3D lighting and geometry bottleneck on PCs. GLINT Gamma implements the full 3D lighting and geometry pipeline for any 3Dlabs rendering device, e.g. GLINT 500TX, GLINT MX, etc. The lighting and geometry calculations in GLINT Gamma are general purpose and may be used to accelerate any 3D API, including OpenGL, Direct3D and Apple's QuickDraw 3D.

The GLINT Gamma2 contains two on-chip PCI Local Bus interfaces: the primary interface communicates with the host processor and the secondary interface communicates with other PCI devices such as GLINT 500TX, GLINT MX or an SVGA device.

GLINT Gamma functions as a AGP/PCI to PCI multi-function adapter. So, in addition to calculating the geometric information, the GLINT Gamma can act as a bridge between the PCI bus and multiple graphics devices. This capability may be used in various ways:

- Driving twin GLINT 500TX or GLINT MX devices for increased rendering speed;
- Driving a GLINT rendering device plus an SVGA device for 3D acceleration with on-board VGA.

## 1.2 The GLINT Family

The GLINT 500TX and MX graphics processors provide 100% OpenGL compliant rendering combined with state-of-the-art Windows acceleration. VRAM framebuffer support enables the high screen resolutions required by professional applications such as CAD and visualization.

The GLINT Gamma is completely compatible with GLINT 500TX and MX rendering devices providing a glueless hardware interface. The GLINT Gamma programming model is fully compatible with the other GLINT devices.

# 2

# Functional Overview

The GLINT Gamma adds extra 3D graphics acceleration to the 3Dlabs' rendering devices by implementing the geometry, lighting, slope and setup calculations for 3D primitives in hardware. The processing required on the host is greatly reduced and much less data is passed from the host to the graphics subsystem.

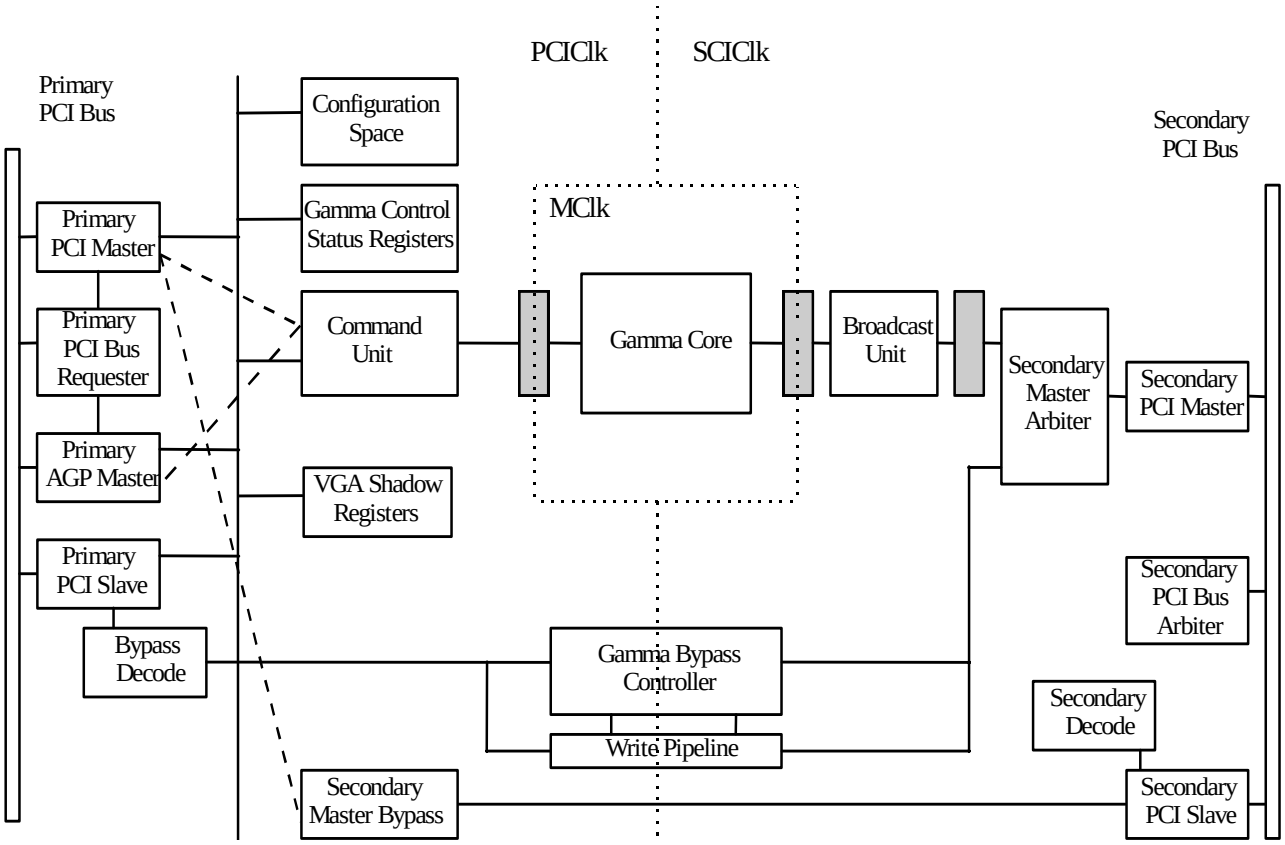


**Figure 2-1 Functional Overview**

For PCI accesses, if the access is for the Graphics Processor, it is directed towards the Command unit. If it is for a secondary target, the access is directed towards the Gamma Bypass Controller. The bus interface contains a number of PCI Configuration Registers and also various Control Status registers for the Gamma.

## 2.1 Reset Mode Control

A number of the parameters for the bus interface are set at reset time. There are two modes of reset configuration control in Gamma. The first mode is Delta compatible, the

second uses a Serial EEPROM (Serial Mode) to allow greater configuration set-up control. In Delta compatible mode there are some hardwired mode pins. In both modes some of the reset state is configured using resistors connected to configuration pins. These pins normally form part of the Gamma operation, but are tri-state at reset. The state of configuration pins is sampled on the trailing edge of reset.

The resulting reset state is stored as Configuration bits within Gamma.

| Configuration Bit | Description |
|---|---|
| PCI66MHzCapable | 1 = 66MHz Capable |
| AGPCapable | 1 = AGP Capable |
| SBACapable | 1 = AGP Sideband Addressing Capable |
| BaseClassZero | 1 = force PCI Base Class Code to be zero |
| PCIMinGnt[7:6] | top two bits of PCI Minimum Grant register |
| PCIMaxLat[7:6] | top two bits of PCI Maximum Latency register |
| ExtDevice1 | 1 = Enable external device 1 |
| ExtDevice2 | 1 = Enable external device 2 |
| ExtDevice3 | 1 = Enable external device 3 |
| ExtVGAMode | 0 = No VGA<br>1 = External device 1 is a VGA device<br>2 = External device 3 is a VGA device<br>3 = No VGA |
| VGANonAlias | See 2.6 VGA Support for definitions of the VGANonAlias |
| VGAControl | See 2.6 VGA Support for definitions of the VGA control bits |
| GlintEn1 | 1 = External Device 1 is a GLINT device<br>0 = External Device 1 is any other device |
| GlintEn2 | 1 = External Device 2 is a GLINT device<br>0 = External Device 2 is any other device |
| MultiGLINTAp | Multi GLINT aperture size control:<br>if GlintEn1 = 1 and GlintEn2 = 1 then<br>00 = 0M DISABLED<br>01 = 16M<br>10 = 32M<br>11 = 64M<br>else<br>DISABLED |
| SubsystemVendorID[15:0] | Subsystem Vendor ID |
| SubsystemID[15:0] | Subsystem ID |

**Table 2-1 Gamma Configuration Bit Definitions**

### 2.1.1 Delta compatible configuration mode

Delta compatible configuration mode is provided to allow a Gamma chip to be placed directly onto current Delta boards without any hardware changes. All new designs should use the Gamma serial configuration mode.

In Delta compatible mode, the configuration bits are loaded from mode and configuration pins, in exactly the same way as Delta. Some configuration bits are hard-coded and cannot be used in this mode of operation. To switch to Serial configuration mode the ExtFunc pins are set to 00b. This setting would never be used on any current Delta board design, so making the new mode backwards compatible with current Delta boards.

The configuration pins are listed in the tables below:

| Mode Pin | Description | Mode |
|---|---|---|
| ExtFunc[1:0] | 00b | Serial Configuration |
| | 01b<br>ExtDevice1 = 1<br>ExtDevice2 = 0<br>ExtDevice3 = 0 | Delta Compatible |
| | 10b<br>ExtDevice1 = 0<br>ExtDevice2 = 1<br>ExtDevice3 = 0 | Delta Compatible |
| | 11b<br>ExtDevice1 = 1<br>ExtDevice2 = 1<br>ExtDevice3 = 0 | Delta Compatible |

**Table 2-2 External Function Enable Pins (Mode control for Gamma)**

| Mode Pin | Description |
|---|---|
| AGPSBA[0]<br>(was ModeCtl[0]) | 1 = Report legacy PCI Base class BaseClassZero = 1b<br>0 = Report PCI 2.1 Base class BaseClassZero = 0b |
| AGPSBA[1]<br>(was ModeCtl[1]) | 1 = PCIMaxLat[7:6] = 00b PCIMinGnt[7:6] = 11b<br>0 = PCIMaxLat[7:6] = 11b PCIMinGnt[7:6] = 11b |
| VGAEnable | 1= External Device 1 is VGA device ExtVGAMode = 01b<br>0 = No VGA present ExtVGAMode = 00b<br>Note: Delta compatibility mode only supports External Device 1 as VGA |

**Table 2-3 Delta compatible Mode Pins**

| Configuration Pin | Description |
|---|---|
| GLINTInDis[0] | Target device 0 GLINT control<br>Pull low (4K7) if GLINT GlintEn1 = 1<br>else Pull High (4K7) GlintEn1 = 0 |
| GLINTInDis[1] | Target device 1 GLINT control<br>Pull low (4K7) if GLINT GlintEn2 = 1<br>else Pull High (4K7) GlintEn2 = 0 |

**Table 2-4 Delta compatible Configuration Pins**

| Configuration Bits | Value | Note |
|---|---|---|
| PCI66MHzCapable | 0b | |
| AGPCapable | 0b | |
| SBACapable | 0b | |
| VGANonAlias | 0b | |
| VGAControl | 0000b | |
| SubsystemVendorID[15:0] | 0000h | 1 |
| SubsystemID[15:0] | 0000h | 1 |

**Table 2-5 Delta compatible Hard coded configuration bits.**

*Note 1: In Delta compatible mode the subsystem registers are Write-once, reset to 0000h.*

### 2.1.2 Serial configuration mode

In serial configuration mode the Gamma configuration bits are set from configuration pins, and from a serial EEPROM. The EEPROM chosen for use with Gamma is the Xicor X84041-3 or X84041-2.7. For more information on this part please refer to the Xicor data sheet. Other serial EEPROMs may be usable if they are 100% compatible with the Xicor part.

| Pin | Description |
|---|---|
| SerialCEN | Serial EEPROM Chip Enable |
| SerialOEN | Serial EEPROM Output Enable |
| SerialWEN | Serial EEPROM Write Enable |
| SerialData | Serial EEPROM Data bit |

**Table 2-6 Serial Mode Operation: EEPROM Pins**

| Configuration Pin | Description |
|---|---|
| SerialData | Pull High (4k7) if external VGA target present<br>else Pull Low (4k7) |
| SerialOEN:SerialWEN | Secondary PCI Bus Clock Control Set using 4k7 Pull-ups and pull-downs for following codes.<br>00b = SCIClk[0:3] = PCIClk<br>01b = SCIClk[0:3] = PCIClk<br>10b = SCIClk[0:3] = PCIClk/2<br>11b = SCIClk[0:3] = PCIClk/2 |
| VGAEnable | In serial mode this pin is used to set the value of the 66MHz capable bit in the Configuration Status register.<br>0b = 66MHz capable bit not set<br>1b = 66MHz capable bit set |

**Table 2-7 Serial Mode Operation: Configuration Pins**

| EEPROM Data Bit | Configuration Bits |
|---|---|
| 0 | AGPCapable |
| 1 | SBACapable |
| 2 | BaseClassZero |
| [4:3] | PCIMinGnt[7:6] |
| [6:5] | PCIMaxLat[7:6] |
| 7 | ExtDevice1 |
| 8 | ExtDevice2 |
| 9 | ExtDevice3 |
| [11:10] | ExtVGAMode |
| 12 | VGANonAlias |
| [16:13] | VGAControl |
| 17 | GlintEn1 |
| 18 | GlintEn2 |
| [20:19] | MultiGLINTAp[1:0] |
| [47:32] | SubsystemVendorID[15:0] |
| [63:48] | SubsystemID[15:0] |

**Table 2-8 Serial Mode Operation: EEPROM Bit Definitions**

*Note* *The ExtVGAMode bits are qualified by the configuration state on the SerialData pin. If the hardware configuration is set such that VGA target is defined to be absent, the EEPROM bits will be ignored, and ExtVGAMode will be set to 00b.*

*Bits 0 to 20 of the Gamma configuration are visible in the* ***ChipConfig*** *register. This register can be read back over the PCI bus. Some of the bits*

*are writeable and may be modified by the host processor if required — see **ChipConfig** register specification in this document. The PCI66MHzCapable, and subsystem Vendor ID, and subsystem ID are visible only in the appropriate PCI configuration register.*

## 2.2 Gamma multi-function support

The GLINT Gamma includes circuitry to allow multiple single function PCI devices attached to its secondary bus to look like a single PCI multi-function device. To achieve this the Gamma device adjusts the results of certain secondary device configuration space accesses.

### 2.2.1 PCI Multi-function indication

To indicate a multi-function device all functions must return Bit 7 of the Header Type register set to a '1'. This bit is set by Gamma during a read of a secondary device Header Type register.

### 2.2.2 PCI 66MHz capable

To indicate 66MHz capability all functions must return Bit 5 of the Header Type register set to a '1'. This bit is set by Gamma during a read of a secondary device Status register if the Gamma PCI66MHzCapable configuration bit is set.

### 2.2.3 Force Legacy Class Codes

If **BaseClassZero** is set, GLINT Gamma will override the PCI base and sub class reported by devices on the secondary PCI bus. GLINT Gamma and any GLINT rendering device on the secondary PCI bus will have 00h reported as their PCI base class and 00h reported as their PCI sub-class. Any VGA device on the secondary PCI bus will have 00h reported as its PCI base class and 01h reported as its PCI sub-class.

## 2.3 PCI Address Regions

The Gamma PCI interface implements four PCI Address Regions, shown in Table 2-2.

The standard VGA-compatible Memory and I/O Space addresses are decoded when the Gamma has been suitably configured. These addresses do not form a single contiguous region, but are mentioned in the table for completeness.

| Region | Address Space | Bytes | Description | Comments |
|---|---|---|---|---|
| Config | Configuration | 256 | PCI Configuration | PCI special |
| Zero | Memory | 128K | Control Registers | relocatable |
| Two | Memory | Note 1. | Multi-GLINT aperture | relocatable |
| ROM | Memory | 0K | Expansion ROM | No ROM |
| VGA | Memory & I/O | — | VGA Addresses | Note 2. |

**Table 2-9 Gamma PCI Address Regions.**

*Note 1: Multi-GLINT aperture size is variable. Size is set by EEPROM configuration bits at reset.*

*Note 2: VGA Addresses are decoded by Gamma when then VGA Present configuration bit is set to be processed by VGA chip on the secondary Bus.*

## 2.4 PCI Configuration Space

The PCI Configuration Space provides a set of 'hooks' which satisfies the needs of current and anticipated system configuration mechanisms. The configuration registers are accessed and modified by the use of PCI Configuration Read and Write commands, and will normally be initialized by BIOS or similar low-level code at system power-up and reset.

Sixty four bytes of the Configuration Registers are predefined within the PCI Specification and are supported by the Gamma. These are defined in Section 28.2 of this document, and are all implemented within the PCI Bus Interface. Registers are provided for device identification, PCI control and status, and as base address registers for the relocatable memory regions. Registers are also provided to allow the reading and writing of the Gamma configuration serial EEPROM.

## 2.5 Gamma Control Registers

The Gamma Region Zero is a 128KByte region containing the control registers, and ports to and from the graphics processor. The control space is mapped in twice within the 128KByte region. In Delta compatible mode the second 64K the registers are mapped to be byte swapped for big endian hosts. For Dual GLINT systems when in Serial mode, Region zero can be configured to allow both GLINT control regions to be visible, the second GLINT being in the second 64K of this region.

A number of Control Status Registers are implemented within the PCI Bus Interface, including registers for interrupt and error handling, reporting graphics processor input FIFO status, and DMA control. All other registers on the Target GLINT are visible through the Gamma Control register space to allow for minimum software changes in supporting a Gamma Device.

## 2.6 VGA Support

The bus interface can be configured to respond to the standard VGA-compatible Memory and I/O Space addresses configuring ExtVGAMode. The interface will then respond to Memory addresses A0000h through BFFFFh.

Gamma can be configured to allow accesses to a number of I/O addresses to allow different types of VGA device to be used on the secondary bus.

### 2.6.1 VGA Aliasing

Aliases of the VGA I/O addresses will be decoded as defined below:

| Decode Type | Address operation | VGANonAlias | VGAControl |
|---|---|---|---|
| 10 Bit aliasing | Bits 31-10 are ignored | 0b | 0b |
| partial aliasing | Bits 31-16 must be 0000h<br>Bits 15-10 are ignored | 0b | 1b |
| Full decode | Bits 31-10 must be 000000h | 1b | xb |

**Table 2-10 VGA Address Aliasing**

### 2.6.2 VGA decode control

The VGA address ranges to be decoded can be configured in Serial mode, as set out in the table below:

| I/O Address range | Type | Configuration Control |
|---|---|---|
| 3B0h - 3BBh, 3C0h - 3DFh | VGA | VGAControl(0)<br>1 = disabled<br>0 = enabled |
| 2E8h, 2EAh - 2EFh | XGA | VGAControl(1)<br>1 = disabled<br>0 = enabled |
| 102h | Special | VGAControl(2)<br>1 = disabled<br>0 = enabled |

**Table 2-11 VGA Address Regions**

In Delta compatible mode **VGAControl**(3-0) is set to 0h and **VGANonAlias** to set to 0b. This enables the decoding of all the I/O address ranges with full 10 bit address aliasing to make Gamma fully compatible with Delta in this mode.

## 2.7 VGA register shadowing

The GLINT Gamma shadows a number of the VGA I/O control registers to allow it to work transparently in all systems. These shadow registers ensure that Gamma only responds to the correct VGA memory addresses, and I/O addresses for the current VGA set-up.

This shadowing allows a Gamma to co-exist with a monochrome adapter which is something that was not possible with a GLINT Delta.

## 2.8 Multi-GLINT aperture

The Gamma Multi-GLINT aperture supports two GLINTS with non-shared framebuffers. Addresses into the aperture consist of a Y and an X component. To simplify calculations the X Span into the aperture is always a power of 2. The size of the Gamma aperture is set at reset.

The Aperture Address generator multiplies the Y address by the GLINT X span and then adds the result to the X address.

GLINT X spans supported are 320, 512, 640, 800, 1024, 1152, 1280, 1600, 1920 in 8 bits, 16 bits and 32 bits per pixel.

The address generator can handle GLINT framebuffer aperture sizes of 4M, 8M, 16M, and 32 Mbytes.

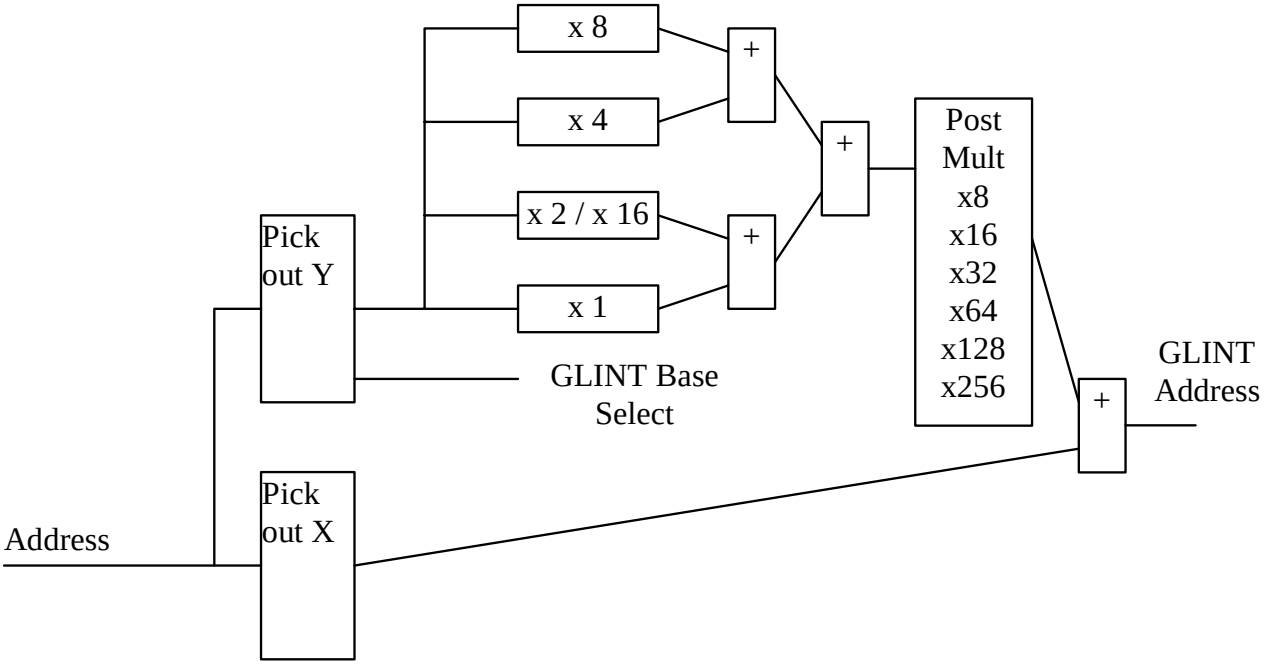


**Figure 2-2 Address generation circuitry**

*Note: The Y address used in forming the GLINT address is ½ of the Input Y address, and the bottom bit of the Input Y address is used to select which GLINT Base address to use in the framebuffer access.*

# 3

# PCI Configuration Region

The PCI Configuration Region provides at set of ‘hooks’ which satisfies the needs of current and anticipated system configuration mechanisms. The configuration registers are accessed and modified by the use of Configuration Read and Write commands.

As GLINT Gamma is a multi-function device, the configuration space is split into eight 256 bytes blocks. GLINT Gamma has one internal function and up to 3 external functions.

GLINT Gamma will only respond to configuration space accesses for which devices exist. In Delta mode, hardware mode pins are used to indicate which of the external functions are populated in the system.

## 3.1 Internal Function Configuration Registers

64 bytes of the Configuration registers are predefined within the PCI Specification and are supported by GLINT Gamma. The remaining 192 Bytes are device specific AGP capabilities are configured in a number of registers in this device specific area. All other registers are unused by GLINT Gamma, returning the value zero.

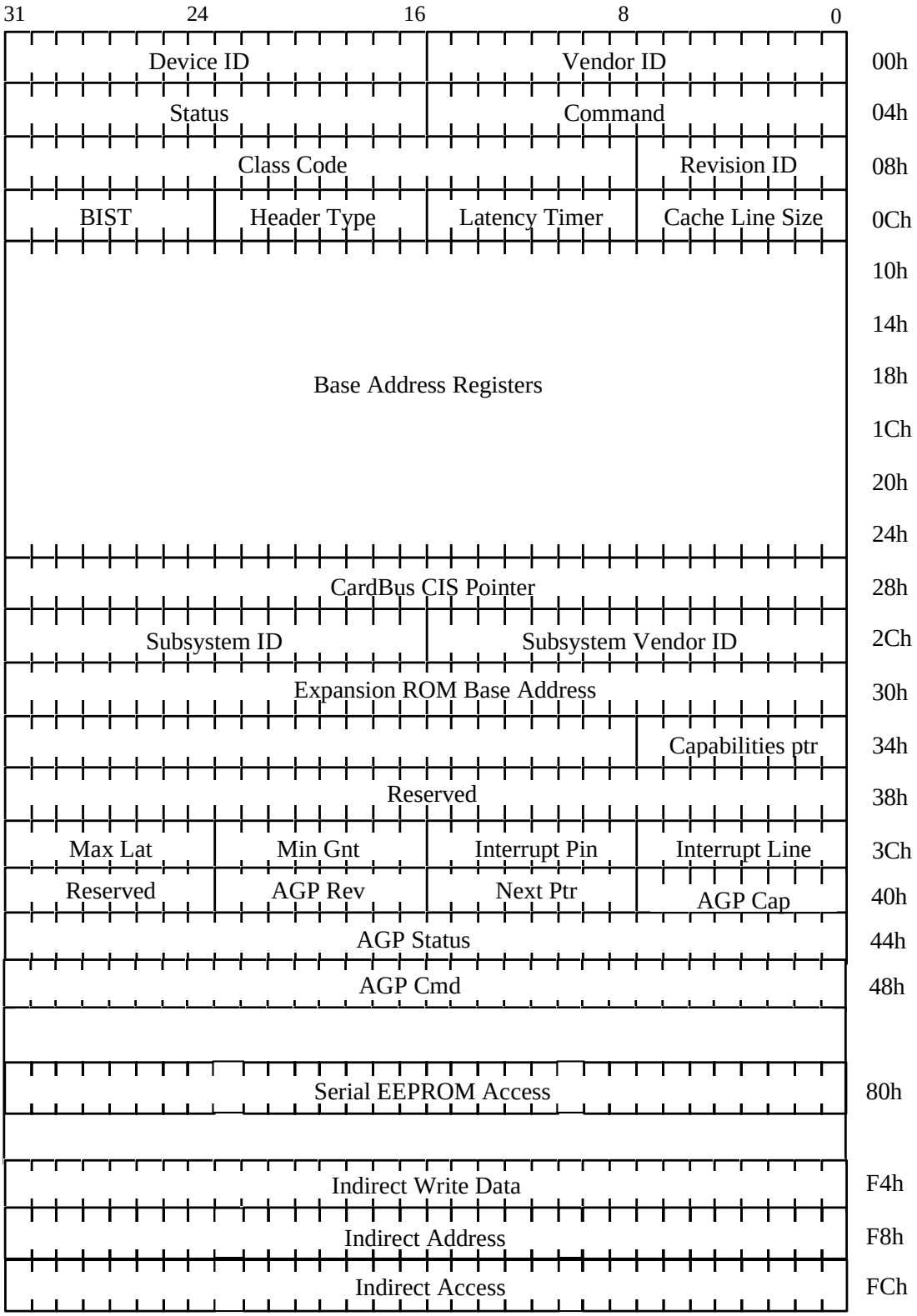


**Figure 3-1 Configuration Region.**

## 3.2 Device Identification

### CFGVendorID

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGVendorID | Config | 0x00 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...15 | Vendor ID | ✔ | ✕ | 0x000E | 3Dlabs company code |

Notes: Vendor Identification numbers

### CFGDeviceID

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGDeviceID | Config | 0x0002 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 16...31 | Device ID | ✔ | ✕ | 0x0008 | GLINT Gamma Device number |

Notes: Device Identification number

### CFGRevisionID

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGRevisionID | Config | 0x08 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...7 | Revision ID | ✔ | ✕ | Rev. no. | GLINT Gamma Device number |

Notes: Revision identification number. The revision ID register returns the following code:
01h = Revision R01

## CFGClassCode

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGRevisionID | Config | 0x09 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 8...15 | Device class | ✔ | ✕ | 0x0B4000 | At offset 0x00 |
| 16...23 | Sub class | ✔ | ✕ | 0x0B4000 | PCI Definition: Co-Processor (unknown type) at offset 0x40 |
| 24...31 | Base class | ✔ | ✕ | 0x0B4000 | PCI Definition: Processor at offset 0x0B |

Notes: Class Code Register Revision identification number. The revision ID register returns the following code: 01h = Revision R01

## CFGHeaderType

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGHeaderType | Config | 0x0e | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 16...23 | HeaderType | ✔ | ✕ | 0x80 | Header Type. PCI Definition: Multi- function device |

Notes:

## 3.3 Device Control

### 3.3.1 Command Register

The command register controls the ability of a device to generate and respond to PCI cycles. Writing zero to this register disconnects the device from the PCI for all except configuration accesses. All necessary bits within the command register are supported for the functionality contained in GLINT Gamma.

## CFGCommand

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGCommand | Config | 0x04 | Bitfield |

***Command register***

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | I/O access enable | ✔ | ✕ | 0x0000 | 0 = GLINT Gamma has no I/O space regions |
| 1 | Memory access enable | ✔ | ✔ | 0x0000 | 0 = Disable memory space accesses. (RESET)<br>1 = Enable memory space accesses |
| 2 | Master enable | ✔ | ✔ | 0x0000 | 0 = Disable master accesses. (RESET)<br>1 = Enable master accesses |
| 3 | Special Cycle access enable | ✔ | ✕ | 0x0000 | 0 = GLINT Gamma has no I/O space regions |
| 4 | Memory Write and Invalidate enable | ✔ | ✕ | 0x0000 | 0 = GLINT Gamma master never issues Memory Write and Invalidate accesses. |
| 5 | VGA palette snoop enable | ✔ | ✕ | 0x0000 | 0 = GLINT Gamma is not a VGA device |
| 6 | Parity error report enable | ✔ | ✕ | 0x0000 | 0 = GLINT Gamma does not report parity errors. |
| 7 | Address/Data stepping enable | ✔ | ✕ | 0x0000 | 0 = GLINT Gamma does not do stepping. |
| 8 | SERR driver enable | ✔ | ✕ | 0x0000 | 0 = GLINT Gamma does not report parity errors. |
| 9 | Master Fast back-to-back enable | ✔ | ✕ | 0x0000 | 0 = The GLINT Gamma master does not implement fast back-to-back accesses. |
| 10...15 | Reserved | ✕ | ✕ | 0x0000 | 000000b |

## 3.4 Device Status

### 3.4.1 Status Register

The Status Register is used to record status information for PCI related events. The definition for each bit is given below.

Reads to this register behave normally. Writes function differently in that bits can be reset but not set. A bit is reset whenever the register is written and the data in the corresponding bit location is a 1.

## CFGStatus

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGStatus | Config | 0x06 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 16...19 | Reserved | ✔ | ✕ | 0x00 | 0.0000b |
| 20 | Cap_List | ✔ | ✕ | | Configured by **AGPCapable**.<br>0 = no additional capabilities over PCI2.1<br>1 = AGP Capability |
| 21 | 66 MHz Capable | ✔ | ✕ | | Configured by 66MHz capable configuration pin<br>0 = Gamma is 33 MHz capable only.<br>1 = Gamma is 66 MHz capable ( in an AGP system) |
| 22 | UDF Supported | ✔ | ✕ | | 0 = Gamma does not support user-definable configurations. |
| 23 | Fast Back-to-Back Capable | ✔ | ✕ | | 1 = Gamma can accept fast back-to-back PCI transactions. |
| 24 | Data Parity Error Detected | ✔ | ✕ | | 0 = Parity checking not implemented on the Gamma. |
| 25-26 | DEVSEL Timing | ✔ | ✕ | | 01b = The Gamma asserts DEVSEL# at medium speed. |
| 27 | Signaled Target Abort | ✔ | ✕ | | 0 = The Gamma never signals Target-Abort. |
| 28 | Received Target Abort | ✔ | ✕ | | This bit is set by the Gamma bus master whenever its transaction is terminated with Target-Abort. |
| 29 | Received Master Abort | ✔ | ✕ | | This bit is set by the Gamma bus master whenever its transaction is terminated with Master-Abort. |
| 30 | Signaled System Error | ✔ | ✕ | | 0 = The Gamma never asserts a system error. |
| 31 | Detected Parity Error | ✔ | ✕ | | 0 = Parity checking is not implemented by the Gamma. |

## 3.5 Miscellaneous Functions

### CFGBist

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGStatus | Config | 0x0f | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 24...31 | BIST | ✔ | ✕ | 0x00 | BIST unsupported by GLINT Gamma over the PCI interface. |

Notes: Optional register used for control and status of BIST (built-in self-test).

### CFGLatTimer

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGLatTimer | Config | 0x0d | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 8...15 | Latency Timer Count | ✔ | ✕ | 0x00 | Sets the maximum number of PCI clock cycles for master burst accesses. |

Notes: This register specifies, in PCI bus clocks, the value of the Latency Timer for this PCI bus master.

### CFGCacheLine

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGCacheLine | Config | 0x0c | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...7 | Cache Line Size | ✔ | ✕ | 0x00 | Cache line size unsupported. |

Notes: This register specifies the cache line size in units of 32 bit words. It is only implemented for masters which use the 'Memory write and invalidate' command. GLINT Gamma does not use this command.

## CFGMaxLat

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGMaxLat | Config<br>***Command register*** | 0x3f | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...5 | Maximum Latency | ✔ | ✕ | 0x80 | 00.0000b |
| 6 | Maximum Latency[6] | ✔ | ✕ | | Set by the bit PCIMaxLat[0] |
| 7 | Maximum Latency[7] | ✔ | ✕ | | Set by the bit PCIMaxLat[1] |

Notes: This register specifies how often the PCI device needs to gain access to the PCI bus. Two EEPROM bits are used to set the top two bits of this register, and the lower bits are always zero. (Possible register values are thus limited to 00h, 40h, 80h, and C0h.)

## CFGMinGrant

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGMinGrant | Config<br>***Command register*** | 0x3e | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...5 | Maximum Grant[5:0] | ✔ | ✕ | configured | 00.0000b |
| 6 | Minimum Grant[6] | | | | Set by the bit PCIMinGnt[0] |
| 7 | Minimum Grant[7] | | | | Set by the bit PCIMinGnt[1] |

Notes: This register specifies how long a burst period the PCI device needs. Two EEPROM bits are used to set the top two bits of this register, and the lower bits are always zero. (Possible register values are thus limited to 00h, 40h, 80h, and C0h.)

## CFGIntPin

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGIntPin | Config<br>***Command register*** | 0x3d | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...7 | IntPin | ✔ | ✕ | 0x01 | 01h GLINT Gamma uses Interrupt pin A |

Notes: The Interrupt Pin register tells the BIOS which interrupt line GLINT Gamma uses

## CFGIntLine

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGIntLine | Config<br>***Command register*** | 0x3c | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...7 | IntLine | ✔ | ✕ | 0x00 | 01h GLINT Gamma uses Interrupt pin A |

Notes: The Interrupt Line register is an 8 bit register used to communicate interrupt line routing information.

## CFGCardBus

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGCardBus | Config<br>***Command register*** | 0x28 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...31 | CardBus pointer | ✔ | ✕ | 0000.0000h | 0000.0000h = Not implemented |

Notes:

## CFGSubsystemVendorId

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGSubsystemVendorId | Config<br>***Command register*** | 0x3c | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...15 | Subsystem Vendor ID | ✔ | ✔ | configured | |

Notes: This register identifies the vendor of the add-in board on which the Gamma device resides.
In Delta Compatible reset mode it has a reset value of zero, and can only be written to once; all subsequent writes are discarded. Both bytes of this register should be initialized by the Gamma BIOS after a reset (write once). In Serial Mode the Subsystem ID is read only and is loaded from the Serial EEPROM.

## CFGSubsystemId

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGSubsystemId | Config<br>***Command register*** | 0x2e | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 16...31 | Subsystem ID | ✔ | ✔ | configured | |

Notes: This register is used to identify the add-in board on which the Gamma device resides.
In Delta Compatible reset mode it has a reset value of zero, and can only be written to once; all subsequent writes are discarded. Both bytes of this register should be initialized by the Gamma BIOS after a reset (write once). In Serial Mode the Subsystem ID is read only and is loaded from the Serial EEPROM.

## CFGCapPtr

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGCapPtr | Config | 0x2e | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...7 | Capability Ptr | ✔ | ✔ | configured | Configured by AGPCapable<br>00h. when AGPCapable = 0<br>40h when AGPCapable = 1 |

Notes: The Capabilities Pointer register is an eight bit register used to provide an offset into the configuration space for the first item in a capabilities list. It is used by the Gamma device in an AGP system to point to the AGP capability registers.

## 3.6 AGP Registers

### CFGCapID

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGCapID | Config | 0x40 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...7 | Capability ID | ✔ | ✕ | configured | Configured by AGPCapable<br>0x00. when AGPCapable = 0<br>0x02 when AGPCapable = 1 |

Notes: This register specifies that the device has AGP capability

### CFGNextPtr

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGNextPtr | Config | 0x41 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 08...15 | Next Ptr | ✔ | ✕ | 0x00 | 00h = no further capabilities in list |

Notes: This register specifies that the device has no next capability item.

## CFGAGPRev

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGAGPRev | Config | 0x42 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 16...19 | Minor Rev | ✔ | ✕ | configured | Configured by AGPCapable<br>0h if AGPCapable = 0<br>0h if AGPCapable = 1 |
| 20...23 | Major Rev | ✔ | ✕ | configured | Configured at reset by AGPCapable<br>00h if AGPCapable = 0<br>01h if AGPCapable = 1 |

Notes: This register specifies the revision of the AGP spec the device conforms to

## CFGAGPStatus

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGAGPStatus | Config | 0x44 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...1 | Rate | ✔ | ✕ | configured | Configured by AGPCapable<br>0h if AGPCapable = 0<br>1h if AGPCapable = 1 |
| 2...8 | Reserved | ✕ | ✕ | | |
| 9 | SBA | | | | Configured by AGPCapable and SBACapable<br>0 if AGPCapable = 0 or SBACapable = 1<br>1 if AGPCapable = 1 and SBACapable =1 |
| 10...23 | Reserved | | | | |
| 24...31 | RQ | | | | Maximum number of AGP requests supported<br>Configured by AGPCapable<br>00h if AGPCapable = 0<br>1Fh if AGPCapable = 1 |

Notes: This register describes the AGP capabilities of the device.

## CFGAGPCommand

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGAGPCommand | Config | 0x48 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...1 | DataRate | ✔ | ✔ | 0x00000000 | |
| 2...7 | Reserved | ✕ | ✕ | | |
| 8 | AGPEnable | ✔ | ✔ | | 0 = AGP Mastering Disabled<br>1 = AGP Mastering Enabled |
| 9 | SBAEnable | ✔ | ✔ | | 0 = sideband Addressing Disabled<br>1 = sideband Addressing Enabled |
| 10...23 | Reserved | ✕ | ✕ | | |
| 24...31 | RQDepth | ✔ | ✔ | | Maximum number of AGP requests which can be queued |

Notes:

# 3.7 Serial EEPROM Access

## CFGAGPCommand

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGAGPCommand | Config | 0x48 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...1 | DataRate | ✔ | ✔ | 0x00000000 | |
| 2...7 | Reserved | ✕ | ✕ | | |
| 8 | AGPEnable | ✔ | ✔ | | 0 = AGP Mastering Disabled<br>1 = AGP Mastering Enabled |
| 9 | SBAEnable | ✔ | ✔ | | 0 = sideband Addressing Disabled<br>1 = sideband Addressing Enabled |
| 10...23 | Reserved | ✕ | ✕ | | |
| 24...31 | RQDepth | ✔ | ✔ | | Maximum number of AGP requests which can be queued |

Notes:

## CFGSerialEnable

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGSerialEnable | Config | 0x80 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0. | Serial EEPROM Data Pin | ✔ | ✔ | 0x00000000 | |
| 1...31 | Reserved | ✕ | ✕ | | |

Notes: This register is used to access the Configuration Serial EEPROM. Reads and writes of the EEPROM can be initiated by setting up the correct stream of read and write accesses to the bottom bit of this register. For details of the read / write patterns to be used, see the XICOR data sheet on the X84041 operation. Note the EEPROM is also programmable and readable through a region 0 register.

# 3.8 Indirect Region 0 access though configuration space

Three configuration registers are provided to allow indirect accesses to region 0 through configuration space accesses. The offset to the region that is to be accessed is loaded into the Indirect Address register. For a write access, the data to be written is loaded into the Indirect Write Data register. The write access is then initiated by writing to the Indirect Access register. The data value written into the Indirect Access register is ignored. For a read access a read of the Indirect Access register returns the values at the current Indirect Address offset.

## CFGIndirectWriteData

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGIndirectWriteData | Config | 0xf4 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...31 | Indirect write data | ✔ | ✔ | 0x00000000 | |

Notes: The Indirect Write Data register holds data value which will be written with Indirect write access.

## CFGIndirectAddress

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGIndirectAddress | Config<br>***Command register*** | 0xf8 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...22 | IndirectAddress | ✔ | ✔ | 0x00000000 | Address offset within region |
| 23...27 | Reserved | ✕ | ✕ | | |
| 28...31 | Base Address select | | | | 0 = Base Address 0<br>1 to 7 = Reserved |

Notes:

## CFGIndirectAccess

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGIndirectAccess | Config<br>***Command register*** | 0xfc | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...31 | Indirect Data | ✔ | ✔ | 0x00000000 | Writes: Data value ignored<br>Reads: Indirect access read data value |

Notes: Accessing the Indirect access register initiates the indirect region 0 access

# 3.9 Base Addresses

The base address registers allow the boot software to relocate PCI devices in the address spaces. At system power-up device independent software must be able to determine which devices are present, build a consistent address map, and determine if a device has an expansion ROM. All undefined Base address registers are read only and return the value 0000.0000h. GLINT Gamma does not have an expansion ROM and so the expansion ROM base address is also read only, returning the value 0000.0000h.

## CFGBaseAddr0

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGBaseAddr0 | Config<br>***Command register*** | 0x10 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...3 | Address Type | ✔ | ✕ | 0x0000.0000 | 0x0 Memory Space, not prefetchable, in 32 bit address space |
| 4...16 | Size indication | ✔ | ✕ | | 0x000 Indicates that the control registers must be mapped into 128 KBytes |
| 17...31 | Base Offset | | | | Loaded at boot time to set offset of the control register space. |

Notes: The Base Address 0 Register contains the GLINT Gamma control space offset. The control registers are in memory space. They are not prefetchable and can be located anywhere in 32 bit address space.

## CFGBaseAddr2

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGBaseAddr2 | Config | 0x18 | Bitfield |

***Command register***

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...3 | Address Type | ✔ | ✕ | 0x0000.0000 | 0h Memory Space, not prefetchable, in 32 bit address space |
| 4...xx | Size Indication | ✔ | ✕ | | These bits are zero, Region size set at configuration to 16M, 32M, 64M or disabled |
| 31...xx | Base Address | ✔ | ✕ | | **Loaded at boot time to set offset of the Multi GLINT aperture** |

Notes: The Base Address 2 Register contains the base address of the Gamma Multi GLINT aperture, and defines the size and type of this region.

## 3.10 Gamma Functions

As a PCI multi-function device Gamma can have up to 4 functions. If Gamma is configured as AGP capable the Gamma function is always the first function. This is because the AGP capable function is always assumed to be the first function in a multi-function device. When not AGP capable the order of the functions is affected by the **ExtVGAMode** mode bits.

| PCI Function | AGP / No VGA | Device 1 VGA | Device 3 VGA |
|---|---|---|---|
| 0 | Gamma Function | External Device 1 | External Device 3 |
| 1 | External Device 1 | Gamma Function | Gamma Function |
| 2 | External Device 2 | External Device 2 | External Device 1 |
| 3 | External Device 3 | External Device 3 | External Device 2 |
| 4 | | | |
| 5 | | | |
| 6 | | | |
| 7 | | | |

**Table 3-1 Gamma PCI Address Regions.**

# 4

# Region 0 - Control Registers

## 4.1 Region 0 Address Map

The Gamma Region Zero is a 128KByte region containing the control registers, and ports to and from the graphics processor. The control space is mapped in twice within the 128KByte region. In Delta compatible mode the second 64K the registers are mapped to be byte swapped for big endian hosts. Region zero can be configured in other ways for dual GLINT systems. See the **CSRAperture** register specification for details.

A large part of this region is not actually Gamma registers, but rather the registers of the GLINT Target device sitting below Gamma. This allows Gamma to be a transparent as possible to the driver software.

| Address Range | Region Select | Register Location |
|---|---|---|
| 0000.0000 -> 0000.0FFF | Control Status | Gamma / Target |
| 0000.1000 -> 0000.1FFF | Target Control | Target |
| 0000.2000 -> 0000.2FFF | GP FIFO Access | Gamma / Target |
| 0000.3000 -> 0000.7FFF | Target Control | Target |
| 0000.8000 -> 0000.FFFF | GP Registers | Gamma / Target |
| 0001.0000 -> 0001.0FFF | Control Status | Gamma / Target |
| 0001.1000 -> 0001.1FFF | Target Control | Target |
| 0001.2000 -> 0001.2FFF | GP FIFO Access | Gamma / Target |
| 0001.3000 -> 0001.7FFF | Target Control | Target |
| 0001.8000 -> 0001.FFFF | GP Registers | Gamma / Target |

**Table 4-1 Region Zero Address Map.**

## 4.2 Control Status Registers

The GLINT Gamma Control Status Register region is split into two sections. The lower section allows direct access to the control status registers of the GLINT rendering device connected to GLINT Gamma. Some of the registers in this section are actually GLINT Gamma registers which are shadowing GLINT 500TX or MX operations for software compatibility with systems without GLINT Gamma.

The upper section has additional GLINT Gamma registers which are documented below. Refer to the appropriate *GLINT 500TX* or *MX Hardware Reference Manual* for details on registers in the lower region.

Writes to any reserved or undefined registers in the Control Status area will be discarded; reads will return the value zero.

For information on the target register definitions please refer to the appropriate target Hardware Reference Manual.

| Address Range | Register | Register Location | Notes |
|---|---|---|---|
| 0000.0000 | ResetStatus | Gamma/Target | 1 |
| 0000.0008 | IntEnable | Target | 2 |
| 0000.0010 | IntFlags | Target | 2 |
| 0000.0018 | InFIFOSpace | Gamma | |
| 0000.0020 | OutFIFOWords | Target | 2 |
| 0000.0028 | DMAAddress | Gamma | |
| 0000.0030 | DMACount | Gamma | |
| 0000.0038 | ErrorFlags | Target | 2 |
| 0000.0040 | VClkCtl | Target | 2 |
| 0000.0048 | TestRegister | Target | 2 |
| 0000.0050 - 0000.0058 | Target Registers | Target | 2 |
| 0000.0060 | DMAControl | Gamma | |
| 0000.0068 | FIFODiscon | Target | 2 |
| 0000.0070 - 0000.07F8 | Target Registers | Target | 2 |
| 0000.0080 | OutDMA | Gamma | |
| 0000.0088 | OutDMACount | Gamma | |
| 0000.0090 - 0000.03F8 | Target Registers | Target | 2 |
| 0000.0400 - 0000.07F8 | Target Registers | Target | 3 |
| 0000.0800 | GResetStatus | Gamma | |
| 0000.0808 | GIntEnable | Gamma | |
| 0000.0810 | GIntFlags | Gamma | |
| 0000.0818 - 0000.0830 | Reserved | Gamma | |
| 0000.0838 | GErrorFlags | Gamma | |
| 0000.0840 | Reserved | Gamma | |
| 0000.0848 | GTestRegister | Gamma | |
| 0000.0850 - 000.0860 | Reserved | Gamma | |
| 0000.0868 | GFIFODis | Gamma | |
| 0000.0870 | GChipConfig | Gamma | |
| 0000.0878 | GCSRAperture | Gamma | |
| 0000.0880 - 0000.0BF8 | Reserved | Gamma | |
| 0000.0C00 | PageTableAddr | Gamma | |
| 0000.0C08 | PageTableLength | Gamma | |
| 0000.0C10 | Reserved | Gamma | |
| 0000.0C18 | Reserved | Gamma | |

| Address Range | Register | Register Location | Notes |
|---|---|---|---|
| 0000.0C20 | Reserved | Gamma | |
| 0000.0C28 | Reserved | Gamma | |
| 0000.0C30 | Reserved | Gamma | |
| 0000.0C38 | DelayTimer | Gamma | |
| 0000.0C40 | CommandMode | Gamma | |
| 0000.0C48 | CommandIntEnable | Gamma | |
| 0000.0C50 | CommandIntFlags | Gamma | |
| 0000.0C58 | CommandErrorFlags | Gamma | |
| 0000.0C60 | CommandStatus | Gamma | |
| 0000.0C68 | CommandFaulting Addr | Gamma | |
| 0000.0C70 | VertexFaultingAddr | Gamma | |
| 0000.0C78 | Reserved | Gamma | |
| 0000.0C80 | Reserved | Gamma | |
| 0000.0C88 | WriteFaultingAddr | Gamma | |
| 0000.0C90 | Reserved | Gamma | |
| 0000.0C98 | FeedbackSelectCount | Gamma | |
| 0000.0CA0 | Reserved | Gamma | |
| 0000.0CA8 | Reserved | Gamma | |
| 0000.0CB0 | Reserved | Gamma | |
| 0000.0CB8 | GammaProcessorMode | Gamma | |
| 0000.0CC0 - 0000.0CF8 | Reserved | Gamma | |
| 0000.0D00 | VGAShadow | Gamma | |
| 0000.0D08 | MultiGLINTAperture | Gamma | |
| 0000.0D10 | MultiGLINT1 | Gamma | |
| 0000.0D18 | MultiGLINT2 | Gamma | |
| 0000.0D20 - 0000.0EF8 | Reserved | Gamma | |
| 0000.0F00 | SerialAccess | Gamma | |
| 0000.0F08 - 0000.0FF8 | Reserved | Gamma | |

**Table 4-2 Control Status Register Address Map**

*Note 1: Writing to this register Resets both the Target device and Gamma. The access is sent to the target and snooped by Gamma*

*Note2: The specification for other Target Registers can be found in the appropriate Hardware Reference Manual*

*Note 3: The address offset range 0400h - 07F8h is passed through to the target device to allow visibility of the GLINT registers overlaid by Gamma registers. This is possible because not all address bits are used in the register space decode of GLINT chips.*

### 4.2.1 Reset Status Register

Writing to the reset status register forces a software reset of the GLINT Gamma Graphics Core. The software reset does not reset the GLINT Gamma primary PCI interface. It is provided for software diagnostics in case an incorrect register set up locks up the GLINT Gamma internal GC.

Various mode pins are sampled at reset, these pins are described in section 2.1.

## DResetStatus

| Name | Type | Offset | Format |
|---|---|---|---|
| DResetStatus | Config | 0x00 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...30 | Reserved | ✗ | ✗ | 0x00000000 | |
| 31 | Software Reset Flag | ✔ | ✔ | | 0 = GLINT Gamma is ready for use<br>1 = GLINT Gamma is being reset and must not be used |

Notes: The software reset takes a number of cycles and the GC must not be used during the reset. A flag in the register is provided which shows that the software reset is still in progress

## InFIFOSpace

| Name | Type | Offset | Format |
|---|---|---|---|
| CFGSerialEnable | Config | 0x18 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...31 | Input FIFO Space | ✔ | ✗ | 0x0000.0021 | The number of empty words in the input FIFO. This number of words can be written before checking "InFIFOSpace" again. |

Notes: The Input FIFO Space register shows the number of words that can currently be written to the input FIFO. This register can be read at any time.

### 4.2.2 DMA Start Address

When using the legacy DMA controller to load the graphics processor, the DMA Start Address register should be loaded with the PCI address of the first word in the buffer to be transferred. In Queued DMA mode writes to this register are ignored.

Writing to the DMA Start Address register loads the address into the DMA address counter. Once a DMA has been initiated, the next DMA start address may be loaded. A read of this register returns the last start value loaded even if the DMA is already underway.

## DMAAddress

| Name | Type | Offset | Format |
|---|---|---|---|
| DMAAddress | Config<br>***Command register*** | 0x28 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...31 | DMA Start Address | ✔ | ✔ | 0x0000.0000 | PCI start address for PCI master read transfer to the graphics processor core. |

Notes:

## DMACount

| Name | Type | Offset | Format |
|---|---|---|---|
| DMACount | Config<br>***Command register*** | 0x30 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...23 | DMA Count | ✔ | ✔ | 0x0000.0000 | Number of words to be transferred in the DMA operation |
| 24...31 | Reserved | ✕ | ✕ | | |

Notes: When using the legacy DMA controller to load the graphics processor, the DMA Count register should be loaded with the number of words to be transferred in the DMA operation. In Queued DMA mode writes to this register are ignored.
The action of loading a word greater than zero initiates the DMA operation. The value read back from this register indicates the current number of words left to be transferred. Writes to this register will be ignored if a DMA is in progress, but can be read at any time.

## DMAControl

| Name | Type | Offset | Format |
|---|---|---|---|
| DMAControl | Region 0 | 0x60 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0. | InDMA Byte Swap Control | ✔ | ✔ | 0x0000.0000 | This field should only be changed when the InDMA controller is idle.<br>0 = Standard.<br>1 = Byte Swapped. |
| 1 | InDMA using AGP | ✔ | ✔ | | 0 = DMA uses PCI Master<br>1 = DMA uses AGP Master. |
| 2 | InDMA Data Throttle | ✔ | ✔ | | 0 = use AGP RBF# and IRDY# to throttle data.<br>1 = Only request data when space is available |
| 3 | AGP Long Read Disable | ✔ | ✔ | | 0 = AGP Long Read Requests may be generated.<br>1 = AGP Long Read Requests disabled. |
| 4 | OutDMA Byte Swap Control | ✔ | ✔ | | This field should only be changed when the OutDMA controller is idle.<br>0 = Standard.<br>1 = Byte Swapped. |
| 5 | AGP Data Throttle | ✔ | ✔ | | Applies to all AGP transfers<br>0 = Control data flow using bus protocols.<br>1 = Throttle Data requests based on FIFO space. |
| 6 | AGP High Priority | ✔ | ✔ | | Applies to all AGP transferees<br>0 = Use AGP low priority reads.<br>1 = Use AGP high priority reads. |
| 7...31 | Reserved | ✕ | ✕ | | |

Notes: The DMA control register sets up the data transfer modes for the DMA controllers. Data transfer can be set to byte swapped for big endian hosts.

## OutDMAStartAddress

| Name | Type | Offset | Format |
|---|---|---|---|
| OutDMAStartAddress | Region 0 | 0x80 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...31 | Out DMA Start Address | ✔ | ✔ | 0x0000.0000 | PCI start address for PCI master write transfer from the graphics processor core output FIFO |

Notes: When using the Out DMA controller to upload the graphics processor output FIFO, the Out DMA Start Address register should be loaded with the PCI address where the first word of the upload data is to be transferred.
Writing to the OutDMA Start Address register loads the address into the OutDMA address counter. Once a DMA has been initiated, the next DMA start address may be loaded. A read of this register returns the last start value loaded even if the DMA is already underway.

## OutDMACount

| Name | Type | Offset | Format |
|---|---|---|---|
| OutDMACount | Region 0<br>***Command register*** | 0x88 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...23 | DMA Count | ✔ | ✔ | 0x0000.0000 | Number of words to be transferred in the DMA operation |
| 24...31 | Reserved | ✕ | ✕ | | |

Notes: The Out DMA Count register should be loaded with the number of words to be transferred in the DMA operation. The action of loading a word greater than zero initiates the DMA operation. The value read back from this register indicates the current number of words left to be transferred. Writes to this register will be ignored if it is non-zero, but can be read at any time.

## GResetStatus

| Name | Type | Offset | Format |
|---|---|---|---|
| GResetStatus | Region 0<br>***Control register*** | 0x800 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...30 | Reserved | ✕ | ✕ | | |
| 31 | Software Reset Flag | ✔ | ✔ | 0x0000.0000 | 0 = Gamma is ready for use.<br>1 = Gamma is being reset and must not be used. |

| | |
|---|---|
| Notes: | Writing to the reset status register causes a software reset of Gamma. The software reset does not reset the bus interface. The reset takes a number of cycles to complete during which the graphics processor should not be used. A flag in the register shows that the software reset is still in progress.<br>The GResetStatus register is different to the ResetStatus register in that only the Gamma device is reset, not the Target Device below Gamma. This register is normally for hardware diagnostic purposes only. The software operation is undefined if the Gamma is reset without resetting the Target Device. |

## GIntEnable

| Name | Type | Offset | Format |
|---|---|---|---|
| GIntEnable | Region 0 | 0x808 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | DMA Interrupt Enable | ✔ | ✔ | 0x0000.0000 | 0 = Disable interrupt.<br>1 = Enable interrupt. |
| 1-2 | Reserved | ✕ | ✕ | | |
| 3 | Error Interrupt Enable | ✔ | ✔ | | 0 = Disable interrupt.<br>1 = Enable interrupt. |
| 4...12 | Reserved | ✕ | ✕ | | |
| 13 | Command Interrupt Enable | ✔ | ✔ | | 0 = Disable interrupt.<br>1 = Enable interrupt. |
| 14...31 | Reserved | ✕ | ✕ | | |

Notes: The Interrupt Enable Register selects which internal conditions are permitted to generate a bus interrupt. Three interrupt sources are defined below. At reset all interrupt sources are disabled.

## GIntFlags

| Name | Type | Offset | Format |
|---|---|---|---|
| GIntFlags | Region 0<br>***Control register*** | 0x810 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | DMA Flag | ✔ | ✔ | 0x0000.0000 | 0 = No interrupt.<br>1 = Interrupt outstanding. |
| 1...2 | Reserved | ✕ | ✕ | | |
| 3 | Error Flag | ✔ | ✔ | | 0 = No interrupt.<br>1 = Interrupt outstanding. |
| 4...12 | Reserved | ✕ | ✕ | | |
| 13 | Command Interrupt Flag | ✔ | ✔ | | Flag set when any bits set in the Command Interrupt Flags register<br>0 = No interrupt.<br>1 = Interrupt outstanding. |
| 14...31 | Reserved | ✕ | ✕ | | |

Notes: The Interrupt Flags Register shows which interrupts are outstanding. Flag bits are reset by writing to this register with the corresponding bit set to a one. Flags at positions where the bits are set to zero will be unaffected by the write. The exception to this is bit 13, which is the Command interrupt Flag. This bit is cleared by clearing all bits the Command Unit Interrupt Flags Register **CommandIntFlags.**

## GErrorFlags

| Name | Type | Offset | Format |
|---|---|---|---|
| GErrorFlags | Region 0<br>***Command register*** | 0x838 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | Input FIFO Error Flag | ✔ | ✔ | 0x0000.0000 | Number of words to be transferred in the DMA operation |
| 1....2 | Reserved | ✕ | ✕ | | |
| 3 | DMA Error Flag | ✔ | ✔ | | Flag set for direct or register access to input FIFO while DMA<br>is in progress (i.e. when the DMACount register is not zero).<br>0 = No error.<br>1 = Error outstanding. |

| | | | | | |
|---|---|---|---|---|---|
| 4...6 | Reserved | ✗ | ✗ | | |
| 7 | PCI Master Error Flag | ✔ | ✔ | | Flag set when either Master abort or Target abort occurs while PCI Master access in progress. - The CFGStatus register can be read to determine the type of error.<br>0 = No error.<br>1 = Error outstanding. |
| 8 | Reserved | ✗ | ✗ | | |
| 9 | In DMA Overwrite Error Flag | ✔ | ✔ | | Flag set in legacy mode if the InDMACount register is written when it is not zero<br>0 = No error.<br>1 = Error outstanding. |
| 10 | Out DMA Overwrite Error Flag | ✔ | ✔ | | Flag set in legacy mode if the OutDMACount register is written when it is not zero<br>0 = No error.<br>1 = Error outstanding. |
| 11...12 | Reserved | ✗ | ✗ | | |
| 13 | Command Error Flag | | | | Flag set when any bits set in the Command Error Flags register<br>0 = No error.<br>1 = Error outstanding |
| 14...31 | Reserved | ✗ | ✗ | | |

Notes: The Error Flags register shows which errors are outstanding in Gamma. Flag bits are reset by writing to this register with the corresponding bit set to a one. Flags at positions where the bits are set to zero will be unaffected by the write. The exception to this is bit 13, which is the Command Error Flag. This bit is cleared by clearing all bits the Command Unit Error Flags Register.

## GTestRegister

| Name | Type | Offset | Format |
|---|---|---|---|
| GTestRegister | Region 0 | 0x848 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...23 | DMA Count | ✔ | ✔ | 0x0000.0000 | Number of words to be transferred in the DMA operation |
| 24...31 | Reserved | ✗ | ✗ | | |

Notes: For hardware test purposes only. Setting a bit in this register causes the corresponding unit to be put into test mode.

## DFIFODis

| Name | Type | Offset | Format |
|---|---|---|---|
| DFIFODis | Region 0<br>***Command register*** | 0x868 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0. | Input FIFO Disconnect enable | ✔ | ✔ | 0x0000.0000 | 0 = Disabled (RESET)<br>1 = Enabled |
| 1...31 | Reserved | ✕ | ✕ | | |

Notes: The FIFO Disconnect Register enables input FIFO disconnect on GLINT Gamma. Disconnect is disabled at reset.

## GChipConfig

| Name | Type | Offset | Format |
|---|---|---|---|
| GChipConfig | Region 0<br>***Command register*** | 0x870 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | AGPCapable | ✔ | ✔ | configured | 0 = AGP Capable<br>1 = AGP Capable |
| 1 | SBACapable | ✔ | ✔ | | 0 = AGP sideband Addressing Disable<br>1 = AGP sideband Addressing Enable |
| 2 | BaseClassZero | ✔ | ✔ | | 0 = use the correct PCI Base Class Code<br>1 = force PCI Base Class Code to be zero |
| 3...4 | MinGrant[7:6] | ✔ | ✕ | | Top 2 bits of the PCI Minimum Grant Configuration Register |
| 5...6 | MaxLat[7:6] | ✔ | ✕ | | Specifies how often the PCI device needs to gain access to the PCI bus. |
| 7 | ExtDevice1 | ✔ | ✕ | | Enable External device 1 |
| 8 | ExtDevice2 | ✔ | ✕ | | Enable External device2 |
| 9 | ExtDevice3 | ✔ | ✕ | | Enable External device3 |
| 10...11 | ExtVGAMode | ✔ | ✔ | | 0 = No VGA<br>1 = External Device 1 is a VGA device<br>2 = External Device 3 is a VGA device<br>3 = No VGA |

| | | | | | |
|---|---|---|---|---|---|
| 12 | VGANonAlias | ✔ | ✔ | | 0 = enable VGA 10 bit address aliased decoding<br>1 = disable VGA 10 bit address aliased decoding (Do 32bit decode) |
| 13...16 | VGAControl | ✔ | ✔ | | |
| 17 | GlintEn1 | ✔ | ✕ | | 1 = External Device 1 is a GLINT device<br>0 = External Device 1 is any other device |
| 18 | GlintEn2 | ✔ | ✕ | | 1 = External Device 2 is a GLINT device<br>0 = External Device 2 is any other device |
| 19...20 | Multi GLINTAp | ✔ | ✕ | | Indicates the MultiGLINT aperture size.<br>0 = 0 MBytes.<br>1 = 16 MBytes<br>2 = 32 MBytes<br>3 = 64 MBytes |
| 21...31 | Reserved | ✕ | ✕ | | |

Notes: Most of Gamma configuration bits are visible through the GChipConfig. This register can then be read back over the PCI bus, to allow the host to determine how the Gamma chip has been configured, and to modify various fields of the configuration if required.

*Note: If both GlintEn1 and GlintEn2 are not set then these bits are ignored and the Multi-GLINT aperture size is set to 0MBytes.*

## CSRAperture

| Name | Type | Offset | Format |
|---|---|---|---|
| CSRAperture | Region 0<br>***Command register*** | 0x878 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | CSRView DualGLINT | ✔ | ✔ | 0x0000.0000 | 0 = Low CSR 64K accesses Master GLINT Little Endian<br>High CSR 64K accesses Master GLINT Big Endian<br>1 = Low CSR 64K accesses Master GLINT<br>High CSR 64K accesses Secondary GLINT Endian set by CSRByteSwap |
| 1 | CSRByteSwap | ✔ | ✔ | | This control bit has no effect unless the CSRViewDualGLINT bit is set<br>0 = All CSR accesses are little endian<br>1 = All CSR accesses are big endian |

| | | | | | |
|---|---|---|---|---|---|
| 2 | CSRGLINT Select | ✔ | ✔ | | This control bit has no effect except in a dual GLINT system as defined by the Gamma configuration bits.<br>0 = The Device 1 GLINT is the Master<br>1 = The Device 2 GLINT is the Master |
| 3...31 | Reserved | ✕ | ✕ | | All bits zero |

Notes: The Gamma CSR Aperture Control Register allows the accesses to Gamma and the GLINT chip behind to be configured in a number of different ways for a dual GLINT system. The register is reset to the Delta compatible configuration.

*Note this register has no effect on the region 0 space unless Gamma is controlling a dual GLINT system.*

## PageTableAddr

| Name | Type | Offset | Format |
|---|---|---|---|
| PageTableAddr | Region 0<br>***Command register*** | 0xC00 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...11 | Reserved | ✕ | ✕ | 0x0000.0000 | |
| 12...31 | Page Table Address | ✕ | ✕ | | |

Notes: The Page Table Address register holds the physical address of the page table used during the logical to physical mapping. The base address is given as a byte address, but must be on a 4K byte boundary.

### 4.2.3 Page Table Length Register

The Page Table Length register holds the length, in units of 4K bytes, of the page table. It is only used for some basic range checking.

## PageTableLength

| Name | Type | Offset | Format |
|---|---|---|---|
| PageTableLength | Region 0 | 0xC08 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...9 | Page Table Length | ✔ | ✔ | 0x0000.0000 | Number of words to be transferred in the DMA operation |
| 10...31 | Reserved | ✕ | ✕ | | |

Notes: When using the legacy DMA controller to load the graphics processor, the DMA Count register should be loaded with the number of words to be transferred in the DMA operation. In Queued DMA mode writes to this register are ignored.

The action of loading a word greater than zero initiates the DMA operation. The value read back from this register indicates the current number of words left to be transferred. Writes to this register will be ignored if a DMA is in progress, but can be read at any time.

## DelayTimer

| Name | Type | Offset | Format |
|---|---|---|---|
| DelayTimer | Region 0 | 0xC38 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...31 | Delay Timer Count | ✔ | ✔ | 0x0000.0000 | |

Notes: The Delay Timer Register, when written to, starts a timer. When the timer reaches one an interrupt is generated and the timer stops. Writing zero aborts the timer with no interrupt being generated. Reading this register returns the current timer value.

## CommandMode

| Name | Type | Offset | Format |
|---|---|---|---|
| CommandMode | Region 0 | 0xC40 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...1 | Command Unit Operation | ✔ | ✔ | 0x0000.0000 | 0 = Legacy DMA operation<br>1 = Queued DMA operation<br>2 = Reserved<br>3 = Reserved |
| 2 | Logical addressing | ✔ | ✔ | | 0 = Disable logical addressing<br>1 = Enable logical addressing |
| 3 | Abort Output DMA | ✔ | ✔ | | 0 = Normal operation<br>1 = Abort Output DMA |
| 4...5 | Reserved | ✕ | ✕ | | |
| 6 | Abort Input DMA | ✔ | ✔ | | 0 = Normal operation<br>1 = Abort Input DMA |
| 7...31 | Reserved | ✕ | ✕ | | |

Notes: The Command Mode Register controls the operation of the Command Unit.

## CommandIntEnable

| Name | Type | Offset | Format |
|---|---|---|---|
| CommandIntEnable | Region 0 | 0xC48 | field |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | FIFO Queued Command DMA Interrupt Enable | ✔ | ✔ | 0x0000.0000 | 0 = Interrupt disabled<br>1 = Interrupt enabled |
| 1 | Output DMA Interrupt Enable | ✔ | ✔ | | 0 = Interrupt disabled<br>1 = Interrupt enabled |
| 2 | Command Interrupt Enable | ✔ | ✔ | | 0 = Interrupt disabled<br>1 = Interrupt enabled |

| | | | | | |
|---|---|---|---|---|---|
| 3 | Timer Interrupt Enable | ✔ | ✔ | | |
| 4 | Command Error Interrupt Enable | ✔ | ✔ | | 0 = Interrupt disabled<br>1 = Interrupt enabled |
| 5...7 | Reserved | ✕ | ✕ | | |
| 8 | Page Fault Command Interrupt Enable | ✔ | ✔ | | 0 = Interrupt disabled<br>1 = Interrupt enabled |
| 9 | Page Fault Vertex Interrupt Enable | ✔ | ✔ | | 0 = Interrupt disabled<br>1 = Interrupt enabled |
| 10...11 | Reserved | ✕ | ✕ | | |
| 12 | Page Fault Write Interrupt Enable | ✔ | ✔ | | 0 = Interrupt disabled<br>1 = Interrupt enabled |
| 13...31 | Reserved | ✕ | ✕ | | |

Notes: The Command Interrupt Enable Register selects which command conditions are permitted to generate a bus interrupt. At reset all interrupt sources are disabled.

## CommandIntFlags

| Name | Type | Offset | Format |
|---|---|---|---|
| CommandIntFlags | Region 0 | 0xC50 | Bitfield |

***Command register***

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | FIFO Queued Command DMA Flag | ✔ | ✔ | 0x0000.0000 | 0 = No flag<br>1 = flag |
| 1 | Output DMA Flag | ✔ | ✔ | | 0 = No flag<br>1 = flag |
| 2 | Command Flag | ✔ | ✔ | | 0 = No flag<br>1 = flag |
| 3 | Timer Flag | ✔ | ✔ | | 0 = No flag<br>1 = flag |
| 4 | Command Error Flag | ✔ | ✔ | | 0 = No flag<br>1 = flag |
| 5...7 | Reserved | ✕ | ✕ | | |
| 8 | Page Fault Command Flag | ✔ | ✔ | | 0 = No flag<br>1 = flag |

| | | | | | |
|---|---|---|---|---|---|
| 9 | Page Fault Vertex Flag | ✔ | ✔ | | 0 = No flag<br>1 = flag |
| 10....11 | Reserved | ✕ | ✕ | | |
| 12 | Page Fault Write Flag | ✔ | ✔ | | 0 = No flag<br>1 = flag |
| 13....31 | Reserved | ✕ | ✕ | | |

Notes: Shows which Command interrupts are outstanding. Flag bits are reset by writing to this register with the corresponding bit set to a one. Flags at positions where the bits are set to zero will be unaffected by the write.

## CommandErrorFlags

| Name | Type | Offset | Format |
|---|---|---|---|
| CommandErrorFlags | Region 0 | 0xC58 | Bitfield |

***Command register***

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | Stack underflow error | ✔ | ✔ | 0x0000.0000 | 0 = No error<br>1 = Error |
| 1 | Stack overflow error | ✔ | ✔ | | 0 = No error<br>1 = Error |
| 2 | DMA overrun error | ✔ | ✔ | | 0 = No error<br>1 = Error |
| 3 | Reserved | ✕ | ✕ | | |
| 4 | Page Mapping Fault Command error | ✔ | ✔ | | 0 = No error<br>1 = Error |
| 5 | Page Mapping Fault Vertex error | ✔ | ✔ | | 0 = No error<br>1 = Error |
| 6...7 | Reserved | ✕ | ✕ | | |
| 8 | Page Mapping Fault Write error | ✔ | ✔ | | 0 = No error<br>1 = Error |
| 9 | Reserved | ✕ | ✕ | | |
| 10 | Page Fault Read Access Command error | ✔ | ✔ | | 0 = No error<br>1 = Error |
| 11 | Page Fault Read Access Vertex error | ✔ | ✔ | | 0 = No error<br>1 = Error |
| 12...19 | Reserved | ✕ | ✕ | | |

| | | | | | |
|---|---|---|---|---|---|
| 20 | Page Fault Write Access Write error | ✔ | ✔ | | 0 = No error<br>1 = Error |
| 21 | Reserved | ✕ | ✕ | | |
| 22 | Illegal DMA Tag error | ✔ | ✔ | | 0 = No error<br>1 = Error |
| 23..31 | Reserved | ✕ | ✕ | | |

Notes: The Command Error Flags register shows which errors are outstanding in the Gamma Command Unit. Flag bits are reset by writing to this register with the corresponding bit set to a one. Flags at positions where the bits are set to zero will be unaffected by the write.

## CommandStatus

| Name | Type | Offset | Format |
|---|---|---|---|
| CommandStatus | Region 0 | 0xC60 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | Command DMA busy | ✔ | ✔ | 0x0000.0004 | 0 = Idle<br>1 = Busy |
| 1 | Output DMA Busy | ✔ | ✔ | | 0 = Idle<br>1 = Busy |
| 2 | Input FIFO Empty | ✔ | ✔ | | 0 = Not empty<br>1 = Empty |
| 3..31 | Reserved | ✕ | ✕ | | |

Notes: The Command Status register is a read only register which gives information about the current Command unit status.

## CommandFaultingAddr

| Name | Type | Offset | Format |
|---|---|---|---|
| CommandFaultingAddr | Region 0 | 0xC6 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...31 | Command Faulting Address | ✔ | ✕ | 0x0000.0000 | |

| Notes: | The Command Faulting Address register hold the logical address of an Input DMA access which caused a Command Fault interrupt, or caused one of the three possible logical to physical mapping errors. |
|---|---|

## VertexFaultingAddr

| Name | Type | Offset | Format |
|---|---|---|---|
| VertexFaultingAddr | Region 0 | 0xC70 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...31 | Vertex Faulting Address | ✔ | ✔ | 0x0000.0000 | Number of words to be transferred in the DMA operation |

| Notes: | Vertex Faulting Address register hold the logical address of a Rectangle Read DMA access which caused a Vertex Fault interrupt, or caused one of the three possible logical to physical mapping errors. |
|---|---|

## WriteFaultingAddr

| Name | Type | Offset | Format |
|---|---|---|---|
| WriteFaultingAddr | Region 0 | 0xC88 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...31 | Write Faulting Address | ✔ | ✔ | 0x0000.0000 | Number of words to be transferred in the DMA operation |

| Notes: | The Write Faulting Address register hold the logical address of an Write DMA access which caused a Write Fault interrupt, or caused one of the three possible logical to physical mapping errors. |
|---|---|

.

## FeedbackSelectCount

| Name | Type | Offset | Format |
|---|---|---|---|
| FeedbackSelectCount | Config<br>***Command register*** | 0x30 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...23 | Feedback Select Count | ✔ | ✕ | 0x0000.0000 | |
| 24...30 | Reserved | ✕ | ✕ | | |
| 31 | Feedback Select Count Overflow | ✔ | ✕ | | 0 = No overflow<br>1 = Overflow |

Notes: The operation of this register is currently defined in the Command Unit Specification

### 4.2.4 Gamma Processor Mode Register

## GammaProcesorMode

| Name | Type | Offset | Format |
|---|---|---|---|
| GammaProcesorMode | Region 0<br>***Command register*** | 0x30 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | Delta Enable | ✔ | ✔ | 0x0000.0000 | 0 = Delta unit enabled<br>1 = Delta unit disabled |
| 1..31 | Reserved | ✕ | ✕ | | |

Notes: The Gamma Processor Mode register allows the user to disable the Delta unit in Gamma. This is to allow a target chip Delta unit to be used instead.

## VGAShadow

| Name | Type | Offset | Format |
|---|---|---|---|
| VGAShadow | Region 0<br>***Command register*** | 0xD00 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | VGA IO Color decode | ✔ | ✔ | 0x0000.0001 | 1 = Decode IO color space 3DXh<br>0 = Decode VGA monochrome space 3BXh |
| 1 | VGA Memory enable | | | | 0 = Disable VGA memory<br>1 = Enable VGA memory |
| 2...3 | VGA Memory region Select | | | | 00b = A0000h - BFFFFh (128K)<br>01b = A0000h - AFFFFh (64K)<br>10b = B0000h - B7FFFh (32K)<br>11b = B8000h - BFFFFh (32K) |
| 4...31 | Reserved | ✕ | ✕ | | |

Notes: The VGA shadow control register allows the readback, and also writing of the VGA IO space shadow bits. This allows the VGA shadowing to be set without the use of IO space accesses.

## MultiGLINTAperture

| Name | Type | Offset | Format |
|---|---|---|---|
| MultiGLINTAperture | Region 0<br>***Command register*** | 0xD08 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...1 | Input X Span | ✔ | ✔ | 0x0000.0000 | 00b = Input X span is 1Kbytes<br>01b = Input X span is 2Kbytes<br>10b = Input X span is 4Kbytes<br>11b = Input X span is 8Kbytes |
| 2 | Product Select | ✔ | ✔ | | 0 = Use 2x product<br>1 = Use 16x product (This is used for X Spans of 800 and 1600) |
| 3 | 1x Product Enable | ✔ | ✔ | | |
| 4 | 16x / 2x Product Enable | ✔ | ✔ | | |
| 5 | 4x Product Enable | ✔ | ✔ | | |
| 6 | 8x Product Enable | ✔ | ✔ | | |
| 7...9 | Post Multiply | ✔ | ✔ | | 000b = x8.<br>001b = x16.<br>010b = x32.<br>011b = x64.<br>100b = x128.<br>101b = x256<br>All other values = x8 |
| 10...11 | GLINT Aperture Size | ✔ | ✔ | | 00b = 4M.<br>01b = 8M.<br>10b = 16M.<br>11b = 32M |
| 12.31 | Reserved | ✕ | ✕ | | |

Notes: The Multi-GLINT aperture control register controls the operation of the Multi-GLINT aperture. The control bits define X spans and Y multiply controls to allow the use of non shared framebuffer GLINT systems behind Gamma.

## MultiGLINT1

| Name | Type | Offset | Format |
|---|---|---|---|
| MultiGLINT1 | Region 0<br>***Command register*** | 0xD10 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...21 | Reserved | ✗ | ✗ | 0x0000.0000 | |
| 22..31 | GLINT 1 Base Address | ✔ | ✔ | | |

Notes: The Multi GLINT Base 1 Register should be loaded with the framebuffer base address of the First GLINT in a multi-GLINT system which uses the multi-GLINT aperture.

## MultiGLINT2

| Name | Type | Offset | Format |
|---|---|---|---|
| MultiGLINT2 | Region 0<br>***Command register*** | 0xD18 | Bitfield |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0...21 | Reserved | ✗ | ✗ | 0x0000.0000 | |
| 22..31 | GLINT 2 Base Address | ✔ | ✔ | | |

Notes: The Multi-GLINT Base 2 Register should be loaded with the framebuffer base address of the Second GLINT in a multi-GLINT system which uses the multi-GLINT aperture.

## SerialAccess

| Name | Type | Offset | Format |
|---|---|---|---|
| SerialAccess | Region 0 | 0xF00 | Bitfield |
| | ***Command register*** | | |

| Bits | Name | Read | Write | Reset | Description |
|---|---|---|---|---|---|
| 0 | Serial EEPROM Data Pin | ✔ | ✔ | 0x0000.0000 | |
| 1..31 | Reserved | ✕ | ✕ | | |

Notes: This register is used to access the Configuration Serial EEPROM. Reads and writes of the EEPROM can be initiated by setting up the correct stream of read and write accesses to the bottom bit of this register. For details of the read / write patterns to be used, see the XICOR data sheet on the X84041 operation. Note the EEPROM is also programmable and readable through a configuration register.

## 4.3 Graphics Core Registers

All the Graphics Core registers in the GLINT Gamma, GLINT 500TX and MX are addressed in this part of region 0. The address for each register and associated data fields is defined in the appropriate Programmers Reference Manual.

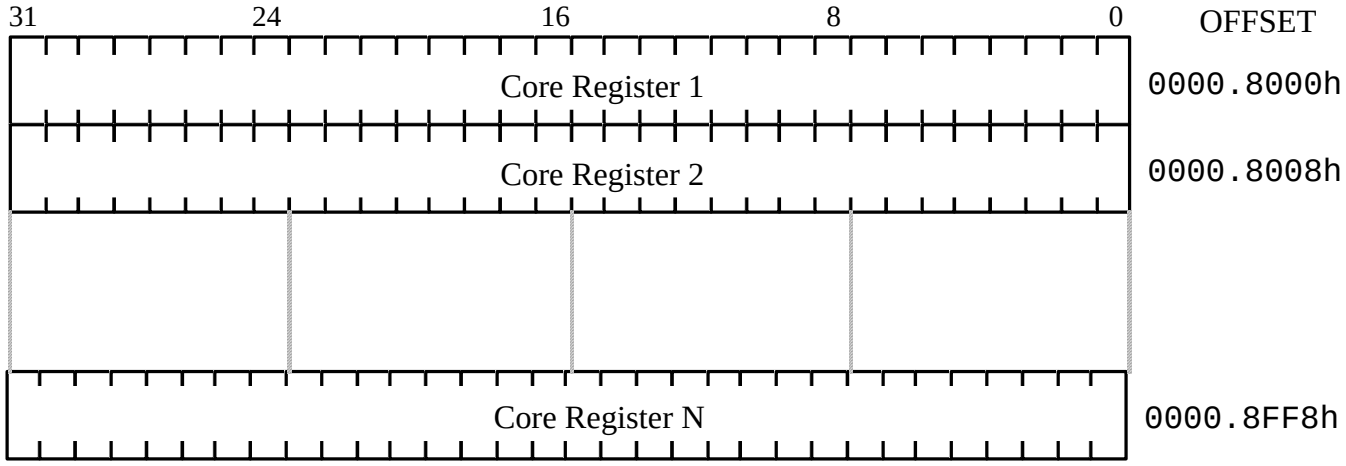


**Figure 4-1 Graphics Core Registers**

*Note: Not all the available register locations are used within the Graphics Core. The registers are on 64 bit boundaries*

## 4.4 Graphics Core FIFO Interface

The Graphics Core FIFO interface provides a port through which both GC register addresses and data can be sent to the input FIFO. A range of 4 KBytes of host space is provided although all data may be sent through one address in the range. ALL accesses go directly to the FIFO, the range is provided to allow for data transfer schemes which force the use of incrementing addresses. Before writing to the input FIFO the user must check that there is sufficient space by reading the InFIFOSpace register.

If the FIFO interface is used, then data is typically sent to the GLINT Gamma in pairs, an address word which addresses the register to be updated, followed by the data to be sent to the register.

*Note: The GC registers can not be read through this interface. Command buffers generated to be sent to the input FIFO interface may be read directly by the GLINT Gamma by using the DMA controller.*

A data formatting scheme is provided to allow for multiple data words to be sent with one address word where adjacent or grouped registers are being written, or where one register is to be written many times.

For more information on the direct FIFO interface data buffer formats please refer to the GLINT Gamma, GLINT 500TX and MX Programmers Reference Manuals.

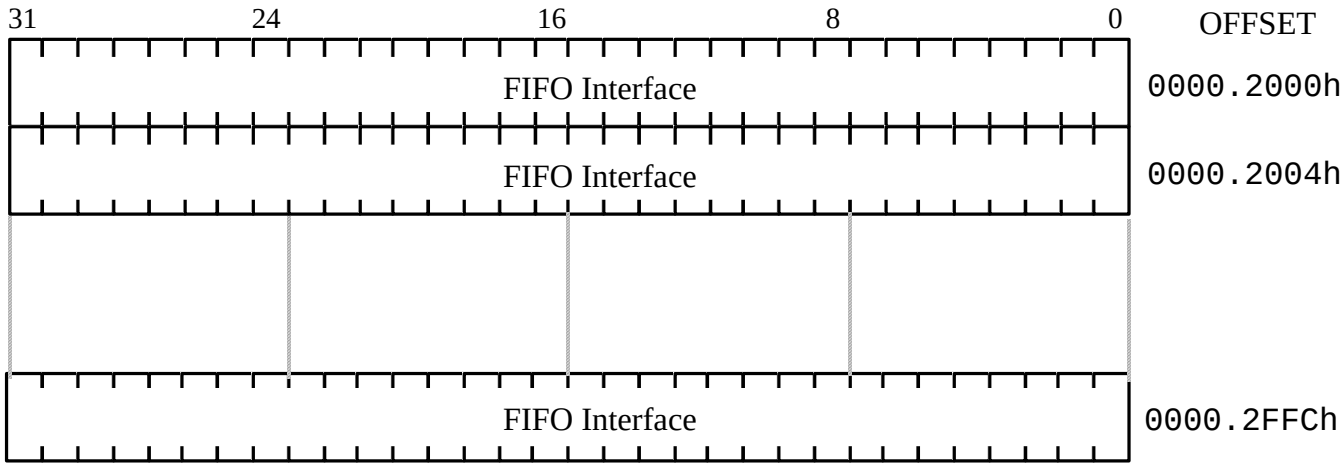


**Figure 4-2 Graphics Core FIFO Interface**

*Note: The FIFO interface can be accessed at 32 bit boundaries. This is to allow a direct copy from a DMA format buffer.*

# 5 Electrical Data

## 5.1 Absolute Maximum Ratings

| | |
|---|---|
| Junction Temperature | 100°C |
| Storage Temperature | -65°C to 150°C |
| I/O DC Supply Voltage (VDD2)<br>Core DC Supply Voltage (VDD) | 3.6V<br>2.62V |
| I/O Pin Voltage with respect to GND | -0.5V to 3.6V |

**Table 5-1 Absolute Maximum Ratings**

## 5.2 DC Specifications

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| VDD2<br>VDD | I/O Supply Voltage<br>Core Supply Voltage | 3.0<br>2.35 | 3.6<br>2.62 | V |
| $L_{PIN}$ | Pin Inductance | | | nH |
| $I_{CC}$ | Power Supply Current | | | mA |

**Table 5-2 DC Specifications**

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $V_{PIL}$ | Input Low Voltage | | 0.8 | V |
| $V_{PIH}$ | Input High Voltage | 2.0 | | V |
| $V_{POL}$ | Output Low Voltage | | 0.5 | V |
| $V_{POH}$ | Output High Voltage | 2.4 | | V |
| $I_{PIL}$ | Input Low Current | | Note 1 | mA |
| $I_{PIH}$ | Input High Current | | Note 1 | mA |
| $C_{PIN}$ | Input Capacitance | | 10 | pF |

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| $C_{CLK}$ | PCI/AGP Clock Input Capacitance | 5 | 12 | pF |
| $C_{IDSEL}$ | PCI Idsel Input Capacitance | | 8 | pF |

**Table 5-3 PCI Signal DC Specifications**

*Note: This value is PCI 2.1 compliant*

## 5.3 AC Specifications

| Pin Name | Capacitive Load |
|---|---|
| PCIAD[31:0], PCICBEN[3:0], PCIPar, PCIFrameN, PCIIRdyN, PCITRdyN, PCIStopN, PCIIdsel, PCIDevselN, PCIReqN, PCIGntN, PCIIntAN, AGPPipeN, AGPRbfN, AGPSBA[7:0], SCIAD[31:0], SCICBEN[3:0], SCIPar, SCIFrameN[1:0],SCIIRdyN[1:0], SCITRdyN[1:0],SCIStopN[1:0], SCIDevselN[1:0]. | 50pFin PCI 33 system. 10pF in AGP system |
| SCIClkOut[3:0],PCIFIFOInDis. | 20pF |

**Table 5-4 Test Loads for AC Timing**

### 5.1.1 Clock Timing

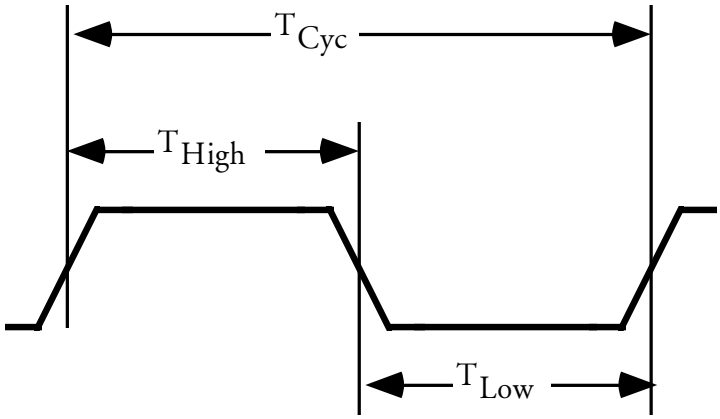


**Figure 5-1 Clock Waveform Timing**

| Symbol | Parameter | Min | Max | Units | Notes |
|---|---|---|---|---|---|
| $T_{PCyc}$ | PCI/AGPClk Cycle Time | 15 | | ns | |
| $T_{PHigh}$ | PCI/AGPClk High Time | 6 | | ns | |
| $T_{PLow}$ | PCI/AGPClk Low Time | 6 | | ns | |
| $T_{MCyc}$ | MClk Cycle Time | 9.25 | | ns | |
| $T_{MHigh}$ | MClk High Time | 3.5 | | ns | |
| $T_{MLow}$ | MClk Low Time | 3.5 | | ns | |
| $T_{SCyc}$ | SCIClk Cycle Time | 15 | | ns | |
| $T_{SHigh}$ | SCIClk High Time | 6 | | ns | |
| $T_{SLow}$ | SCIClk Low Time | 6 | | ns | |

**Table 5-5 Clock Waveform Timing**

### 5.1.2 Input / Output Timing

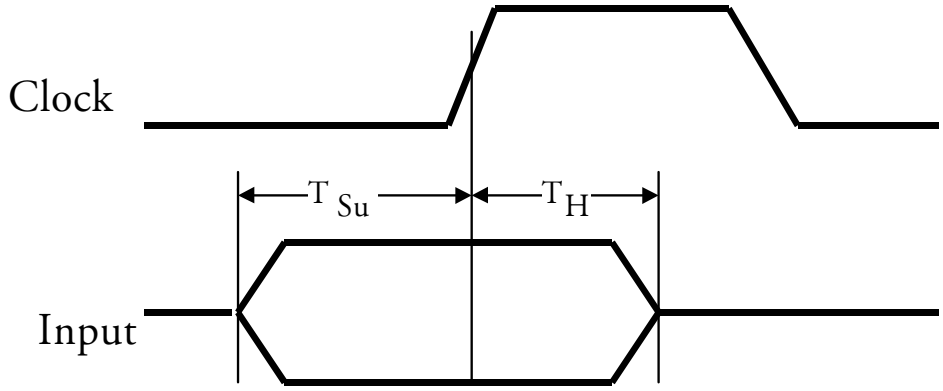


**Figure 5-2 Clock Referenced Input Timing**

| Parameter | $T_{Su}$ Min | $T_H$ Min | Units | Notes |
|---|---|---|---|---|
| PCIAD(31:0), PCICBEN(3:0), PCIPar, PCIFrameN, PCIIRdyN, PCITRdyN, PCIStopN, PCIIdsel, PCIDevselN, AGPSt[2:0] | 5 | 0 | ns | |
| PCIGntN | 5 | 0 | ns | |
| PCIRstN | 7 | 0 | ns | 1 |

**Table 5-6 PCIClk Referenced Input Timing**

*Note 1: PCIRstN is resynchronized internally. The given timings, when met, ensure that the reset is detected in the current cycle.*

| Parameter | TSu Min | TH Min | Units | Notes |
|---|---|---|---|---|
| SCIAD(31:0), SCICBEN(3:0), SCIPar, SCIIRdyN(1:0), SCITRdyN(1:0), SCIStopN(1:0), SCIDevselN(1:0), GLINTInDis(1:0),GLINTOutDis(1:0), SCIReqN | 5 | 0.2 | ns | |
| ModeCtl(1:0), VGAEn, ExtFuncEn(1:0) | 10 | 2 | ns | |

**Table 5-7 SCIClk Referenced Input Timing**

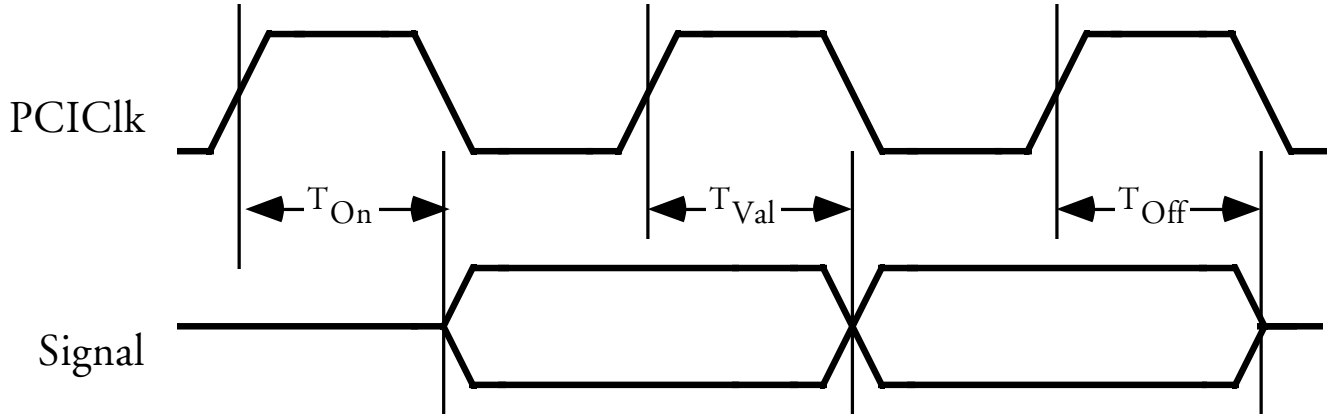


**Figure 5-3 PCIClk Referenced Output Timing**

| | $T_{Val}$ | | $T_{On}$ | | $T_{Off}$ | | | |
|---|---|---|---|---|---|---|---|---|
| **Parameter** | **Min** | **Max** | **Min** | **Max** | **Min** | **Max** | **Units** | **Notes** |
| PCIAD(31:0), PCICBEN(3:0), PCIPar, PCIFrameN, PCIIRdyN, PCITRdyN, PCIStopN, PCIIdsel, PCIDevselN | 2 | 11 | 2 | 11 | 2 | 11 | ns | |
| PCIReqN | 2 | 12 | | | | | ns | |
| PCIIntAN | 2 | 15 | | | | | ns | 1 |

**Table 5-8 PCIClk 33MHz Referenced Output Timing**

*Note 1: Timings given are for falling edges of the open drain signal. Rise times are dependent on the external pull-up resistor.*

| | $T_{Val}$ | | $T_{On}$ | | $T_{Off}$ | | | |
|---|---|---|---|---|---|---|---|---|
| Parameter | Min | Max | Min | Max | Min | Max | Units | Notes |
| PCIAD(31:0), PCICBEN(3:0), PCIPar, PCIFrameN, PCIIRdyN, PCITRdyN, PCIStopN, PCIIdsel, PCIDevselN | 1.5 | 6 | 1.5 | 6 | 2 | 11 | ns | |
| PCIReqN | 1.5 | 6 | | | | | ns | |
| PCIIntAN | 1.5 | 6 | | | | | ns | 1 |

**Table 5-9 PCIClk 66MHz Referenced Output Timing**

*Note 1: Timings given are for falling edges of the open drain signal. Rise times are dependent on the external pull-up resistor.*

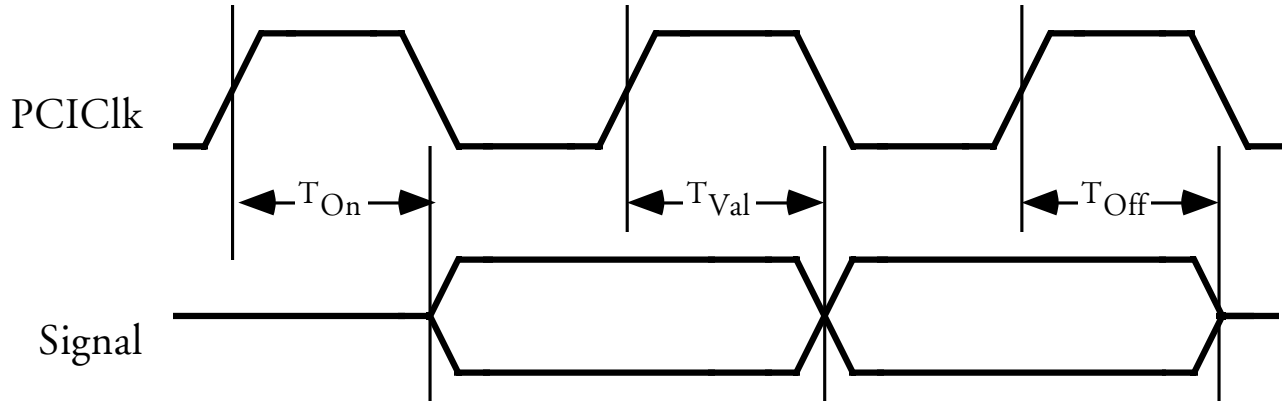


**Figure 5-4 SCIClk Referenced Output Timing**

| | $T_{val}$ | | $T_{On}$ | | $T_{Off}$ | | | |
|---|---|---|---|---|---|---|---|---|
| **Parameter** | **Min** | **Max** | **Min** | **Max** | **Min** | **Max** | **Units** | **Notes** |
| SCIAD(31:0), SCICBEN(3:0), SCIPar, SCIFrameN, SCIIRdyN, SCITRdyN, SCIStopN, SCIDevselN | 1.5 | 7 | 1.5 | 7 | 1.5 | 7 | ns | |
| SCIGntN | 1.5 | 7 | | | | | ns | |

**Table 5-10 SCIClk Referenced Output Timing**

# 6

# Pin Information

## 6.1 Pin Characteristics

The Gamma2 package is a standard 304 ball PQFP part. The pins are shown individually in listings sorted by (a) name and (b) number. Pad number sequence is Counter Clockwise. Pin 1 is on left side of South edge.

| Item | Description |
|---|---|
| Total Package Signal I/O | 228 pins |
| TEST I/O | 64 pins |
| GND | 52 ( 44 Package GNDs ) |
| VDD | 40 ( 16 Package VDDs ) |
| VDD2 | 40 ( 16 Package VDD2s ) |
| N/C | No connection |
| Pads per side | 122 |

**Table 6-1 Pin characteristics**

| PKG PAD | PKG BALL |
|---|---|
| GND | A1, F1, K1, P1, V1, AC1, B2, AB2, C3, AA3, D4, H4, M4, T4, Y4, A6, AC6, D8, Y8, A10, AC10, D12, Y12, A14, AC14, D16, Y16, A18, AC18, D20, H20, M20, T20, Y20, C21, AA21, B22, AB22, A23, F23, K23, P23, V23, AC23 |
| PWR | J4, L4, N4, R4, D9, Y9, D11, Y11, D13, Y13, D15, Y15, J20, L20, N20, R20 |
| PWR2 | E4, G4, U4, W4 |
| PWR3 | Y5, Y7, Y17, Y19 |
| PWR4 | E20, G20, U20, W20 |
| PWR5 | D5, D7, D17, D19 |

**Table 6-2 Ground and Power pin locations**

### 6.1.1 Pinlist by Pin name

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 140 | SIG | 535 | H22 | AGPPipeN | |
| 135 | SIG | 36 | G21 | AGPRbfN | |
| 128 | SIG | 531 | F22 | AGPSBA[2] | |
| 127 | SIG | 34 | E21 | AGPSBA[3] | GRA_Scan_Bypass>OFLAG =BDY;KFLAG=-SG;SP_WAFER_BYPASS=+TI;WRP_DC_BYPASS=+TI |
| 126 | SIG | 33 | E22 | AGPSBA[4] | |
| 119 | SIG | 30 | B21 | AGPSBA[5] | Test_AClk_in>KFLAG=-AC |
| 118 | SIG | 29 | B20 | AGPSBA[6] | Test_BClk_in>KFLAG=-BC |
| 117 | SIG | 28 | B19 | AGPSBA[7] | Test_CClk_in>KFLAG=-SC |
| 139 | SIG | 37 | C22 | AGPSBA_ModeCtl[0] | Boundary_Scan_Enable>KFLAG=+SG |
| 131 | SIG | 35 | F21 | AGPSBA_ModeCtl[1] | |
| 159 | SIG | 41 | J21 | AGPSt[0] | |
| 151 | SIG | 40 | C23 | AGPSt[1] | Unused_test_pad> |
| 144 | SIG | 536 | H21 | AGPSt[2] | |
| 240 | SIG | 60 | Y22 | ExtFuncEn[0] | DI1(0)_NonTest_I/O_driver_inhibit>OFLAG=-BI;KFLAG=-OI;MACRO=-TI;WRP_DC=-TI;SP_WAFER=-TI;WRP_DC_BYPASS=-TI;SP_WAFER_BYPASS=-TI |
| 364 | SIG | 93 | AA4 | ExtFuncEn[1] | DI2(0)_Test_I/O_driver_inhibit>OFLAG=-BI;KFLAG=+SG |
| 427 | SIG | 108 | R1 | GLINTInDis[0] | |
| 439 | SIG | 111 | L1 | GLINTInDis[1] | |
| 447 | SIG | 113 | H1 | GLINTOutDis[0] | |
| 443 | SIG | 112 | J2 | GLINTOutDis[1] | |
| 65 | GND | 16 | A11 | GND | |
| 69 | GND | 17 | A12 | GND | GND> |
| 337 | GND | 85 | AA8 | GND | GND> |
| 321 | GND | 81 | AB9 | GND | GND> |
| 309 | GND | 78 | AC13 | GND | GND> |
| 187 | GND | 47 | L23 | GND | GND> |
| 435 | GND | 110 | M1 | GND | GND> |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 191 | GND | 48 | M23 | GND | GND> |
| 415 | GND | 105 | W1 | GND | GND> |
| 455 | SIG | 115 | E1 | MClk | Clock_input>KFLAG=+SG |
| 312 | SIG | 579 | AA10 | N/C | |
| 304 | SIG | 575 | AA12 | N/C | |
| 300 | SIG | 573 | AA13 | N/C | |
| 241 | SIG | 61 | AA22 | N/C | Unused_pin_and_test_pad> |
| 354 | SIG | 589 | AA6 | N/C | |
| 332 | SIG | 583 | AA9 | N/C | |
| 310 | SIG | 578 | AB10 | N/C | |
| 302 | SIG | 574 | AB12 | N/C | |
| 264 | SIG | 565 | AB16 | N/C | |
| 256 | SIG | 563 | AB17 | N/C | |
| 320 | SIG | 581 | AC9 | N/C | |
| 58 | SIG | 514 | B12 | N/C | |
| 66 | SIG | 518 | B14 | N/C | |
| 10 | SIG | 503 | B7 | N/C | |
| 60 | SIG | 515 | C12 | N/C | |
| 50 | SIG | 511 | D10 | N/C | |
| 114 | SIG | 530 | D18 | N/C | |
| 125 | SIG | 32 | D21 | N/C | |
| 474 | SIG | 618 | F2 | N/C | |
| 130 | SIG | 532 | F20 | N/C | |
| 476 | SIG | 619 | F3 | N/C | |
| 482 | SIG | 620 | F4 | N/C | |
| 132 | SIG | 533 | G22 | N/C | |
| 442 | SIG | 611 | J1 | N/C | |
| 454 | SIG | 613 | J3 | N/C | |
| 432 | SIG | 608 | K2 | N/C | |
| 172 | SIG | 541 | K20 | N/C | |
| 434 | SIG | 609 | K3 | N/C | |
| 436 | SIG | 610 | K4 | N/C | |
| 428 | SIG | 606 | L2 | N/C | |
| 430 | SIG | 607 | L3 | N/C | |
| 424 | SIG | 604 | M2 | N/C | |
| 182 | SIG | 545 | M21 | N/C | |
| 180 | SIG | 544 | M22 | N/C | |
| 426 | SIG | 605 | M3 | N/C | |
| 422 | SIG | 603 | N3 | N/C | |
| 406 | SIG | 599 | P2 | N/C | |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 188 | SIG | 548 | P22 | N/C | |
| 414 | SIG | 600 | P3 | N/C | |
| 386 | SIG | 595 | T2 | N/C | |
| 378 | SIG | 593 | U2 | N/C | |
| 228 | SIG | 557 | U21 | N/C | |
| 372 | SIG | 591 | V2 | N/C | |
| 236 | SIG | 560 | V20 | N/C | |
| 232 | SIG | 559 | V21 | N/C | |
| 230 | SIG | 558 | V22 | N/C | |
| 374 | SIG | 592 | V4 | N/C | |
| 294 | SIG | 571 | Y14 | N/C | |
| 252 | SIG | 562 | Y18 | N/C | |
| 431 | SIG | 109 | N1 | Option | Selects the Gamma clock source: pulled down, the clock is sourced from the MClk pin. Pulled up, it is sourced from the PCI Clock |
| 458 | SIG | 614 | H2 | PCIAD[0] | |
| 459 | SIG | 116 | H3 | PCIAD[1] | |
| 6 | SIG | 501 | B6 | PCIAD[10] | |
| 8 | SIG | 502 | D6 | PCIAD[11] | |
| 13 | SIG | 5 | C7 | PCIAD[12] | |
| 17 | SIG | 6 | B3 | PCIAD[13] | |
| 18 | SIG | 505 | B8 | PCIAD[14] | |
| 25 | SIG | 8 | B4 | PCIAD[15] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 54 | SIG | 512 | B11 | PCIAD[16] | |
| 56 | SIG | 513 | C11 | PCIAD[17] | |
| 57 | SIG | 14 | A8 | PCIAD[18] | |
| 62 | SIG | 516 | B13 | PCIAD[19] | |
| 467 | SIG | 117 | D1 | PCIAD[2] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 64 | SIG | 517 | C13 | PCIAD[20] | |
| 68 | SIG | 519 | C14 | PCIAD[21] | |
| 70 | SIG | 520 | D14 | PCIAD[22] | |
| 77 | SIG | 19 | B15 | PCIAD[23] | |
| 86 | SIG | 523 | C15 | PCIAD[24] | |
| 89 | SIG | 22 | A19 | PCIAD[25] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 90 | SIG | 524 | B16 | PCIAD[26] | |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 101 | SIG | 24 | A20 | PCIAD[27] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 102 | SIG | 526 | B17 | PCIAD[28] | |
| 106 | SIG | 527 | C17 | PCIAD[29] | |
| 468 | SIG | 616 | G2 | PCIAD[3] | |
| 108 | SIG | 528 | B18 | PCIAD[30] | |
| 110 | SIG | 529 | C18 | PCIAD[31] | |
| 472 | SIG | 617 | G3 | PCIAD[4] | |
| 475 | SIG | 119 | B1 | PCIAD[5] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 483 | SIG | 121 | E2 | PCIAD[6] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 484 | SIG | 122 | D2 | PCIAD[7] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 3 | SIG | 1 | C4 | PCIAD[8] | |
| 5 | SIG | 3 | C5 | PCIAD[9] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 485 | SIG | 123 | C2 | PCICBEN[0] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 22 | SIG | 506 | C8 | PCICBEN[1] | |
| 49 | SIG | 12 | A5 | PCICBEN[2] | |
| 74 | SIG | 521 | A15 | PCICBEN[3] | |
| 147 | SIG | 39 | D22 | PCIClk | Clk_input>KFLAG=+SG |
| 38 | SIG | 509 | B10 | PCIDevselN | |
| 46 | SIG | 510 | C10 | PCIFrameN | |
| 148 | SIG | 537 | J22 | PCIGntN | |
| 81 | SIG | 20 | A16 | PCIIdsel | |
| 168 | SIG | 540 | K21 | PCIIntAN | |
| 45 | SIG | 11 | A4 | PCIIRdyN | |
| 26 | SIG | 507 | B9 | PCIPar | |
| 160 | SIG | 539 | K22 | PCIReqN | |
| 167 | SIG | 42 | D23 | PCIRstN | |
| 29 | SIG | 9 | A3 | PCIStopN | Scan_In>OFLAG=BDY;KFLAG=SI |
| 37 | SIG | 10 | C9 | PCITRdyN | |
| 85 | WR | 21 | A17 | N/C | |
| 253 | WR | 66 | AA18 | N/C | |
| 357 | WR | 89 | AA5 | N/C | |
| 329 | WR | 83 | AC7 | N/C | |
| 113 | WR | 27 | C19 | N/C | |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 9 | WR | 4 | C6 | N/C | |
| 479 | WR | 120 | E3 | N/C | |
| 451 | WR | 114 | G1 | N/C | |
| 207 | WR | 52 | U23 | N/C | |
| 375 | WR | 97 | V3 | N/C | |
| 235 | WR | 58 | W21 | N/C | |
| 73 | WR2 | 18 | A13 | N/C | |
| 21 | WR2 | 7 | A2 | N/C | |
| 105 | WR2 | 25 | A21 | N/C | |
| 109 | WR2 | 26 | A22 | N/C | |
| 53 | WR2 | 13 | A7 | N/C | |
| 61 | WR2 | 15 | A9 | N/C | |
| 227 | WR2 | 56 | AA23 | N/C | |
| 387 | WR2 | 100 | AB1 | N/C | |
| 248 | WR2 | 64 | AB19 | N/C | |
| 363 | WR2 | 92 | AB3 | N/C | |
| 313 | WR2 | 79 | AC12 | N/C | |
| 305 | WR2 | 77 | AC15 | N/C | |
| 297 | WR2 | 75 | AC17 | N/C | |
| 265 | WR2 | 69 | AC22 | N/C | |
| 349 | WR2 | 87 | AC3 | N/C | |
| 143 | WR2 | 38 | B23 | N/C | |
| 4 | WR2 | 2 | B5 | N/C | |
| 471 | PWR2 | 118 | C1 | N/C | |
| 93 | WR2 | 23 | C16 | N/C | |
| 120 | WR2 | 31 | C20 | N/C | |
| 486 | WR2 | 124 | D3 | N/C | |
| 175 | WR2 | 44 | G23 | N/C | |
| 183 | WR2 | 46 | J23 | N/C | |
| 195 | WR2 | 49 | N23 | N/C | |
| 423 | WR2 | 107 | T1 | N/C | |
| 215 | WR2 | 54 | T21 | N/C | |
| 419 | WR2 | 106 | U1 | N/C | |
| 242 | WR2 | 62 | Y21 | N/C | |
| 186 | SIG | 547 | N21 | SCIAD[0] | |
| 190 | SIG | 549 | P21 | SCIAD[1] | |
| 231 | SIG | 57 | AB23 | SCIAD[10] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 247 | SIG | 63 | AA20 | SCIAD[11] | |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 249 | SIG | 65 | AA19 | SCIAD[12] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 250 | SIG | 561 | AB18 | SCIAD[13] | |
| 257 | SIG | 67 | AA17 | SCIAD[14] | |
| 261 | SIG | 68 | AB21 | SCIAD[15] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 314 | SIG | 580 | Y10 | SCIAD[16] | |
| 317 | SIG | 80 | AC11 | SCIAD[17] | |
| 325 | SIG | 82 | AC8 | SCIAD[18] | |
| 333 | SIG | 84 | AC5 | SCIAD[19] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 192 | SIG | 550 | P20 | SCIAD[2] | |
| 336 | SIG | 584 | AB8 | SCIAD[20] | |
| 345 | SIG | 86 | AC4 | SCIAD[21] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 346 | SIG | 586 | AB7 | SCIAD[22] | |
| 350 | SIG | 587 | AA7 | SCIAD[23] | |
| 353 | SIG | 88 | AC2 | SCIAD[24] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 360 | SIG | 590 | Y6 | SCIAD[25] | |
| 361 | SIG | 90 | AB5 | SCIAD[26] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 362 | SIG | 91 | AB4 | SCIAD[27] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 369 | SIG | 94 | Y3 | SCIAD[28] | |
| 370 | SIG | 95 | W2 | SCIAD[29] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 196 | SIG | 551 | R23 | SCIAD[3] | |
| 371 | SIG | 96 | W3 | SCIAD[30] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 379 | SIG | 98 | U3 | SCIAD[31] | |
| 199 | SIG | 50 | R22 | SCIAD[4] | |
| 203 | SIG | 51 | T23 | SCIAD[5] | |
| 208 | SIG | 553 | R21 | SCIAD[6] | |
| 211 | SIG | 53 | W23 | SCIAD[7] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 212 | SIG | 554 | T22 | SCIAD[8] | |
| 224 | SIG | 556 | U22 | SCIAD[9] | |
| 223 | SIG | 55 | Y23 | SCICBEN[0] | Unused_test_pad> |
| 266 | SIG | 566 | AA16 | SCICBEN[1] | |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 308 | SIG | 577 | AA11 | SCICBEN[2] | |
| 352 | SIG | 588 | AB6 | SCICBEN[3] | |
| 383 | SIG | 99 | AA2 | SCIClk | Clock_Input>KFLAG=+SG |
| 403 | SIG | 103 | R3 | SCIClkOut[0] | |
| 411 | SIG | 104 | Y1 | SCIClkOut[1] | |
| 416 | SIG | 601 | P4 | SCIClkOut[2] | |
| 420 | SIG | 602 | N2 | SCIClkOut[3] | |
| 281 | SIG | 72 | AA15 | SCIDevselN[0] | |
| 284 | SIG | 569 | AB14 | SCIDevselN[1] | |
| 301 | SIG | 76 | AC16 | SCIFrameN[0] | |
| 306 | SIG | 576 | AB11 | SCIFrameN[1] | |
| 388 | SIG | 596 | T3 | SCIGntN | |
| 391 | SIG | 101 | Y2 | SCIIntAN | NonTest_I/O_receiver_inhibit>KFLAG=-SG;MACRO=-TI;WRP_DC=-TI;SP_WAFER=-TI;WRP_DC_BYPASS=-TI;SP_WAFER_BYPASS=-TI |
| 289 | SIG | 73 | AC20 | SCIIRdyN[0] | |
| 292 | SIG | 570 | AA14 | SCIIRdyN[1] | |
| 269 | SIG | 70 | AB20 | SCIPar | Boundary_Scan_Output_Enable>OFLAG=BDY;IOWRAP=+TI;SP_EXTIO=+TI |
| 395 | SIG | 102 | AA1 | SCIReqN | GRA_Test_CClk2_in>KFLAG=-SC |
| 394 | SIG | 597 | R2 | SCIRstN | |
| 272 | SIG | 567 | AB15 | SCIStopN[0] | |
| 273 | SIG | 71 | AC21 | SCIStopN[1] | Unused_test_pad |
| 293 | SIG | 74 | AC19 | SCITRdyN[0] | |
| 298 | SIG | 572 | AB13 | SCITRdyN[1] | |
| 171 | SIG | 43 | E23 | SerialCEN | |
| 179 | SIG | 45 | H23 | SerialData | |
| 176 | SIG | 542 | L22 | SerialOEN | |
| 178 | SIG | 543 | L21 | SerialWEN | |
| 239 | SIG | 59 | W22 | TestMode | Test_Enable>KFLAG=+TI |
| 184 | SIG | 546 | N22 | VGAEn | |

### 6.1.2 Pin Listing by Ball Coordinates

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 65 | GND | 16 | A11 | GND | |
| 69 | GND | 17 | A12 | GND | GND> |
| 73 | WR2 | 18 | A13 | N/C | |
| 74 | SIG | 521 | A15 | PCICBEN[3] | |
| 81 | SIG | 20 | A16 | PCIIdsel | |
| 85 | WR | 21 | A17 | N/C | |
| 89 | SIG | 22 | A19 | PCIAD[25] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 21 | WR2 | 7 | A2 | N/C | |
| 101 | SIG | 24 | A20 | PCIAD[27] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 105 | WR2 | 25 | A21 | N/C | |
| 109 | WR2 | 26 | A22 | N/C | |
| 29 | SIG | 9 | A3 | PCIStopN | Scan_In>OFLAG=BDY;KFLAG=SI |
| 45 | SIG | 11 | A4 | PCIIRdyN | |
| 49 | SIG | 12 | A5 | PCICBEN[2] | |
| 53 | WR2 | 13 | A7 | N/C | |
| 57 | SIG | 14 | A8 | PCIAD[18] | |
| 61 | WR2 | 15 | A9 | N/C | |
| 395 | SIG | 102 | AA1 | SCIReqN | GRA_Test_CClk2_in>KFLAG=-SC |
| 312 | SIG | 579 | AA10 | N/C | |
| 308 | SIG | 577 | AA11 | SCICBEN[2] | |
| 304 | SIG | 575 | AA12 | N/C | |
| 300 | SIG | 573 | AA13 | N/C | |
| 292 | SIG | 570 | AA14 | SCIIRdyN[1] | |
| 281 | SIG | 72 | AA15 | SCIDevselN[0] | |
| 266 | SIG | 566 | AA16 | SCICBEN[1] | |
| 257 | SIG | 67 | AA17 | SCIAD[14] | |
| 253 | WR | 66 | AA18 | N/C | |
| 249 | SIG | 65 | AA19 | SCIAD[12] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 383 | SIG | 99 | AA2 | SCIClk | Clock_Input>KFLAG=+SG |
| 247 | SIG | 63 | AA20 | SCIAD[11] | |
| 241 | SIG | 61 | AA22 | N/C | Unused_pin_and_test_pad> |
| 227 | WR2 | 56 | AA23 | N/C | |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 364 | SIG | 93 | AA4 | ExtFuncEn[1] | DI2(0)_Test_I/O_driver_inhibit>OFLAG=-BI;KFLAG=+SG |
| 357 | WR | 89 | AA5 | N/C | |
| 354 | SIG | 589 | AA6 | N/C | |
| 350 | SIG | 587 | AA7 | SCIAD[23] | |
| 337 | GND | 85 | AA8 | GND | GND> |
| 332 | SIG | 583 | AA9 | N/C | |
| 387 | WR2 | 100 | AB1 | N/C | |
| 310 | SIG | 578 | AB10 | N/C | |
| 306 | SIG | 576 | AB11 | SCIFrameN[1] | |
| 302 | SIG | 574 | AB12 | N/C | |
| 298 | SIG | 572 | AB13 | SCITRdyN[1] | |
| 284 | SIG | 569 | AB14 | SCIDevselN[1] | |
| 272 | SIG | 567 | AB15 | SCIStopN[0] | |
| 264 | SIG | 565 | AB16 | N/C | |
| 256 | SIG | 563 | AB17 | N/C | |
| 250 | SIG | 561 | AB18 | SCIAD[13] | |
| 248 | WR2 | 64 | AB19 | N/C | |
| 269 | SIG | 70 | AB20 | SCIPar | Boundary_Scan_Output_Enable>OFLAG=BDY;IOWRAP=+TI;SP_EXTIO=+TI |
| 261 | SIG | 68 | AB21 | SCIAD[15] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 231 | SIG | 57 | AB23 | SCIAD[10] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 363 | WR2 | 92 | AB3 | N/C | |
| 362 | SIG | 91 | AB4 | SCIAD[27] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 361 | SIG | 90 | AB5 | SCIAD[26] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 352 | SIG | 588 | AB6 | SCICBEN[3] | |
| 346 | SIG | 586 | AB7 | SCIAD[22] | |
| 336 | SIG | 584 | AB8 | SCIAD[20] | |
| 321 | GND | 81 | AB9 | GND | GND> |
| 317 | SIG | 80 | AC11 | SCIAD[17] | |
| 313 | WR2 | 79 | AC12 | N/C | |
| 309 | GND | 78 | AC13 | GND | GND> |
| 305 | WR2 | 77 | AC15 | N/C | |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 301 | SIG | 76 | AC16 | SCIFrameN[0] | |
| 297 | WR2 | 75 | AC17 | N/C | |
| 293 | SIG | 74 | AC19 | SCITRdyN[0] | |
| 353 | SIG | 88 | AC2 | SCIAD[24] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 289 | SIG | 73 | AC20 | SCIIRdyN[0] | |
| 273 | SIG | 71 | AC21 | SCIStopN[1] | Unused_test_pad |
| 265 | WR2 | 69 | AC22 | N/C | |
| 349 | WR2 | 87 | AC3 | N/C | |
| 345 | SIG | 86 | AC4 | SCIAD[21] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 333 | SIG | 84 | AC5 | SCIAD[19] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 329 | WR | 83 | AC7 | N/C | |
| 325 | SIG | 82 | AC8 | SCIAD[18] | |
| 320 | SIG | 581 | AC9 | N/C | |
| 475 | SIG | 119 | B1 | PCIAD[5] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 38 | SIG | 509 | B10 | PCIDevselN | |
| 54 | SIG | 512 | B11 | PCIAD[16] | |
| 58 | SIG | 514 | B12 | N/C | |
| 62 | SIG | 516 | B13 | PCIAD[19] | |
| 66 | SIG | 518 | B14 | N/C | |
| 77 | SIG | 19 | B15 | PCIAD[23] | |
| 90 | SIG | 524 | B16 | PCIAD[26] | |
| 102 | SIG | 526 | B17 | PCIAD[28] | |
| 108 | SIG | 528 | B18 | PCIAD[30] | |
| 117 | SIG | 28 | B19 | AGPSBA[7] | Test_CClk_in>KFLAG=-SC |
| 118 | SIG | 29 | B20 | AGPSBA[6] | Test_BClk_in>KFLAG=-BC |
| 119 | SIG | 30 | B21 | AGPSBA[5] | Test_AClk_in>KFLAG=-AC |
| 143 | WR2 | 38 | B23 | N/C | |
| 17 | SIG | 6 | B3 | PCIAD[13] | |
| 25 | SIG | 8 | B4 | PCIAD[15] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 4 | WR2 | 2 | B5 | N/C | |
| 6 | Sig | 501 | B6 | PCIAD[10] | |
| 10 | SIG | 503 | B7 | N/C | |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 18 | SIG | 505 | B8 | PCIAD[14] | |
| 26 | SIG | 507 | B9 | PCIPar | |
| 471 | PWR2 | 118 | C1 | N/C | |
| 46 | SIG | 510 | C10 | PCIFrameN | |
| 56 | SIG | 513 | C11 | PCIAD[17] | |
| 60 | SIG | 515 | C12 | N/C | |
| 64 | SIG | 517 | C13 | PCIAD[20] | |
| 68 | SIG | 519 | C14 | PCIAD[21] | |
| 86 | SIG | 523 | C15 | PCIAD[24] | |
| 93 | WR2 | 23 | C16 | N/C | |
| 106 | SIG | 527 | C17 | PCIAD[29] | |
| 110 | SIG | 529 | C18 | PCIAD[31] | |
| 113 | WR | 27 | C19 | N/C | |
| 485 | SIG | 123 | C2 | PCICBEN[0] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 120 | WR2 | 31 | C20 | N/C | |
| 139 | SIG | 37 | C22 | AGPSBA_ModeCtl[0] | Boundary_Scan_Enable>KFLAG=+SG |
| 151 | SIG | 40 | C23 | AGPSt[1] | Unused_test_pad> |
| 3 | Sig | 1 | C4 | PCIAD[8] | |
| 5 | Sig | 3 | C5 | PCIAD[9] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 9 | WR | 4 | C6 | N/C | |
| 13 | SIG | 5 | C7 | PCIAD[12] | |
| 22 | SIG | 506 | C8 | PCICBEN[1] | |
| 37 | SIG | 10 | C9 | PCITRdyN | |
| 467 | SIG | 117 | D1 | PCIAD[2] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 50 | SIG | 511 | D10 | N/C | |
| 70 | SIG | 520 | D14 | PCIAD[22] | |
| 114 | SIG | 530 | D18 | N/C | |
| 484 | SIG | 122 | D2 | PCIAD[7] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 125 | SIG | 32 | D21 | N/C | |
| 147 | SIG | 39 | D22 | PCIClk | Clk_input>KFLAG=+SG |
| 167 | SIG | 42 | D23 | PCIRstN | |
| 486 | WR2 | 124 | D3 | N/C | |
| 8 | SIG | 502 | D6 | PCIAD[11] | |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 455 | SIG | 115 | E1 | MClk | Clock_input>KFLAG=+SG |
| 483 | SIG | 121 | E2 | PCIAD[6] | Scan_In>OFLAG=BDY;KFLAG=SI |
| 127 | SIG | 34 | E21 | AGPSBA[3] | GRA_Scan_Bypass>OFLAG=BDY;KFLAG=-SG;SP_WAFER_BYPASS=+TI;WRP_DC_BYPASS=+TI |
| 126 | SIG | 33 | E22 | AGPSBA[4] | |
| 171 | SIG | 43 | E23 | SerialCEN | |
| 479 | WR | 120 | E3 | N/C | |
| 474 | SIG | 618 | F2 | N/C | |
| 130 | SIG | 532 | F20 | N/C | |
| 131 | SIG | 35 | F21 | AGPSBA_ModeCtl[1] | |
| 128 | SIG | 531 | F22 | AGPSBA[2] | |
| 476 | SIG | 619 | F3 | N/C | |
| 482 | SIG | 620 | F4 | N/C | |
| 451 | WR | 114 | G1 | N/C | |
| 468 | SIG | 616 | G2 | PCIAD[3] | |
| 135 | SIG | 36 | G21 | AGPRbfN | |
| 132 | SIG | 533 | G22 | N/C | |
| 175 | WR2 | 44 | G23 | N/C | |
| 472 | SIG | 617 | G3 | PCIAD[4] | |
| 447 | SIG | 113 | H1 | GLINTOutDis[0] | |
| 458 | SIG | 614 | H2 | PCIAD[0] | |
| 144 | SIG | 536 | H21 | AGPSt[2] | |
| 140 | SIG | 535 | H22 | AGPPipeN | |
| 179 | SIG | 45 | H23 | SerialData | |
| 459 | SIG | 116 | H3 | PCIAD[1] | |
| 442 | SIG | 611 | J1 | N/C | |
| 443 | SIG | 112 | J2 | GLINTOutDis[1] | |
| 159 | SIG | 41 | J21 | AGPSt[0] | |
| 148 | SIG | 537 | J22 | PCIGntN | |
| 183 | WR2 | 46 | J23 | N/C | |
| 454 | SIG | 613 | J3 | N/C | |
| 432 | SIG | 608 | K2 | N/C | |
| 172 | SIG | 541 | K20 | N/C | |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 168 | SIG | 540 | K21 | PCIIntAN | |
| 160 | SIG | 539 | K22 | PCIReqN | |
| 434 | SIG | 609 | K3 | N/C | |
| 436 | SIG | 610 | K4 | N/C | |
| 439 | SIG | 111 | L1 | GLINTInDis[1] | |
| 428 | SIG | 606 | L2 | N/C | |
| 178 | SIG | 543 | L21 | SerialWEN | |
| 176 | SIG | 542 | L22 | SerialOEN | |
| 187 | GND | 47 | L23 | GND | GND> |
| 430 | SIG | 607 | L3 | N/C | |
| 435 | GND | 110 | M1 | GND | GND> |
| 424 | SIG | 604 | M2 | N/C | |
| 182 | SIG | 545 | M21 | N/C | |
| 180 | SIG | 544 | M22 | N/C | |
| 191 | GND | 48 | M23 | GND | GND> |
| 426 | SIG | 605 | M3 | N/C | |
| 431 | SIG | 109 | N1 | Option | Selects the Gamma clock source: pulled down, the clock is sourced from the MClk pin. Pulled up, it is sourced from the PCI Clock |
| 420 | SIG | 602 | N2 | SCIClkOut[3] | |
| 186 | SIG | 547 | N21 | SCIAD[0] | |
| 184 | SIG | 546 | N22 | VGAEn | |
| 195 | WR2 | 49 | N23 | N/C | |
| 422 | SIG | 603 | N3 | N/C | |
| 406 | SIG | 599 | P2 | N/C | |
| 192 | SIG | 550 | P20 | SCIAD[2] | |
| 190 | SIG | 549 | P21 | SCIAD[1] | |
| 188 | SIG | 548 | P22 | N/C | |
| 414 | SIG | 600 | P3 | N/C | |
| 416 | SIG | 601 | P4 | SCIClkOut[2] | |
| 427 | SIG | 108 | R1 | GLINTInDis[0] | |
| 394 | SIG | 597 | R2 | SCIRstN | |
| 208 | SIG | 553 | R21 | SCIAD[6] | |
| 199 | SIG | 50 | R22 | SCIAD[4] | |
| 196 | SIG | 551 | R23 | SCIAD[3] | |
| 403 | SIG | 103 | R3 | SCIClkOut[0] | |
| 423 | WR2 | 107 | T1 | N/C | |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 386 | SIG | 595 | T2 | N/C | |
| 215 | WR2 | 54 | T21 | N/C | |
| 212 | SIG | 554 | T22 | SCIAD[8] | |
| 203 | SIG | 51 | T23 | SCIAD[5] | |
| 388 | SIG | 596 | T3 | SCIGntN | |
| 419 | WR2 | 106 | U1 | N/C | |
| 378 | SIG | 593 | U2 | N/C | |
| 228 | SIG | 557 | U21 | N/C | |
| 224 | SIG | 556 | U22 | SCIAD[9] | |
| 207 | WR | 52 | U23 | N/C | |
| 379 | SIG | 98 | U3 | SCIAD[31] | |
| 372 | SIG | 591 | V2 | N/C | |
| 236 | SIG | 560 | V20 | N/C | |
| 232 | SIG | 559 | V21 | N/C | |
| 230 | SIG | 558 | V22 | N/C | |
| 375 | WR | 97 | V3 | N/C | |
| 374 | SIG | 592 | V4 | N/C | |
| 415 | GND | 105 | W1 | GND | GND> |
| 370 | SIG | 95 | W2 | SCIAD[29] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 235 | WR | 58 | W21 | N/C | |
| 239 | SIG | 59 | W22 | TestMode | Test_Enable>KFLAG=+TI |
| 211 | SIG | 53 | W23 | SCIAD[7] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 371 | SIG | 96 | W3 | SCIAD[30] | Scan_Out>OFLAG=BDY;KFLAG=SO |
| 411 | SIG | 104 | Y1 | SCIClkOut[1] | |
| 314 | SIG | 580 | Y10 | SCIAD[16] | |
| 294 | SIG | 571 | Y14 | N/C | |
| 252 | SIG | 562 | Y18 | N/C | |
| 391 | SIG | 101 | Y2 | SCIIntAN | NonTest_I/O_receiver_inhibit>KFLAG=-SG;MACRO=-TI;WRP_DC=-TI;SP_WAFER=-TI;WRP_DC_BYPASS=-TI;SP_WAFER_BYPASS=-TI |

| No. | Function | Pad | Ball Func-tion | Name | Fields |
|---|---|---|---|---|---|
| 242 | WR2 | 62 | Y21 | N/C | |
| 240 | SIG | 60 | Y22 | ExtFuncEn[0] | DI1(0)_NonTest_I/O_driver_inhibit>OFLAG=-BI;KFLAG=-OI;MACRO=-TI;WRP_DC=-TI;SP_WAFER=-TI;WRP_DC_BYPASS=-TI;SP_WAFER_BYPASS=-TI |
| 223 | SIG | 55 | Y23 | SCICBEN[0] | Unused_test_pad> |
| 369 | SIG | 94 | Y3 | SCIAD[28] | |
| 360 | SIG | 590 | Y6 | SCIAD[25] | |

# 7

# Package Dimensions

The package is a super BGA (SBGA) 304 ball configuration. The bottom view is shown below. The chamfered edge corresponds to the A1 ball location.

**Figure 7-1 Package Diagram (Bottom View)**

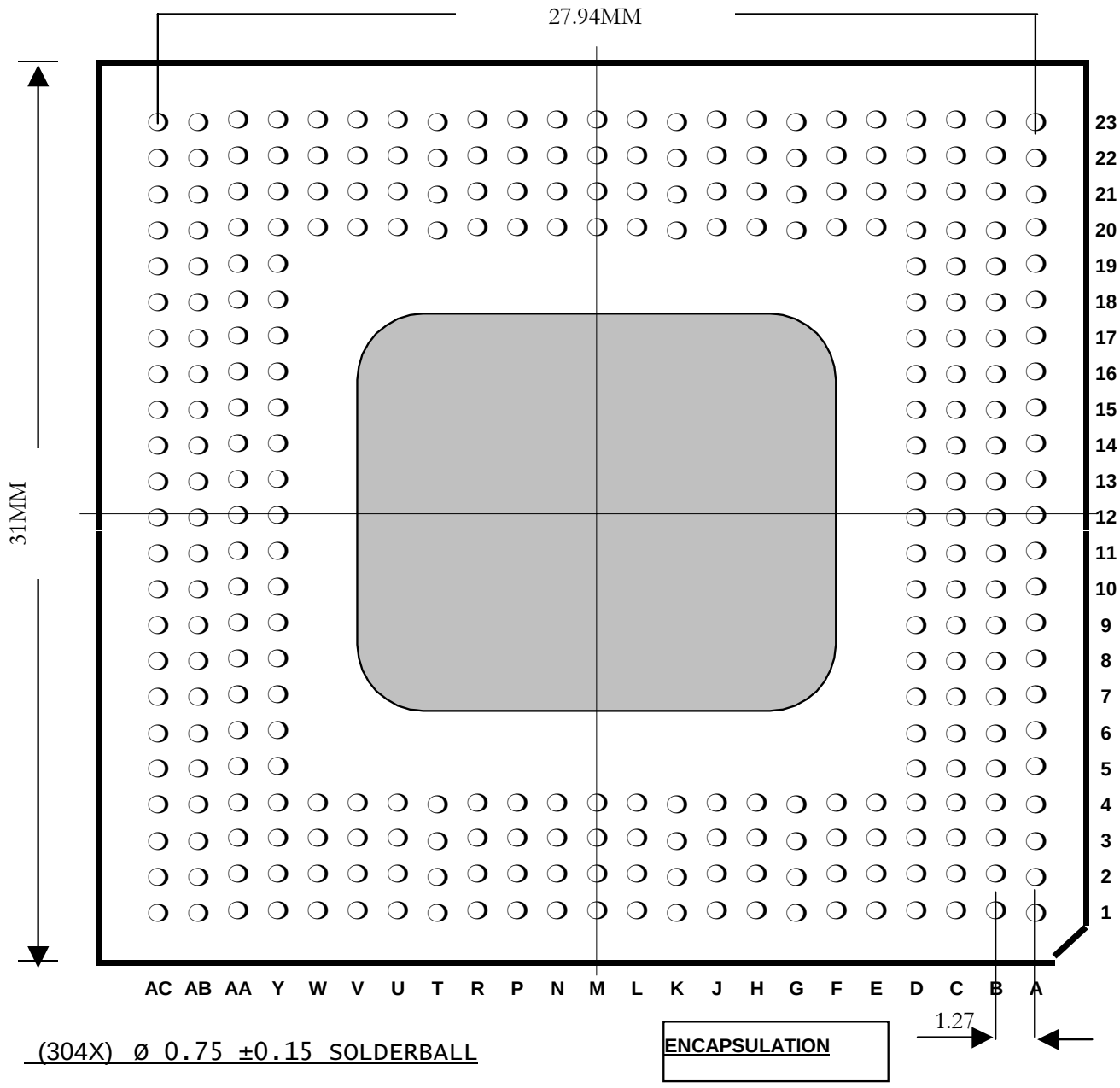

**Figure 7-2 Package Diagram (Top View)**

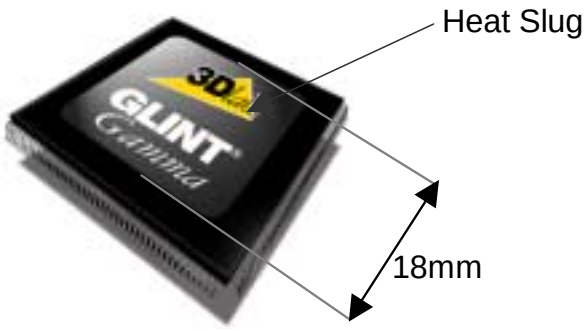


**Figure 7-3 Gamma2 Mechanical Dimensions**

| Dimension | mm |
|---|---|
| Lead Pitch | 1.27 |
| Height (Max) | 2.65 |
| Ball height | 0.60 ± 0.1 |
| Body Width | 31 |
| Encapsultation width | 18 |

**Table 7-4 304 pin SBGA Package Dimensions**

# 8

# Gamma System Design

The following notes should be followed when designing systems which use the Gamma device.

## 8.1 Termination / Pull-ups

The signals SCIFrameN(0:1), SCIIRdyN(0:1), SCITRdyN(0:1), SCIStopN(0:1), SCIDevselN(0:1) are Tri-state signals which need to be held in their inactive state by 10K Ohm pull-up resistors.

SCIIntAN is an open drain signal which should all be pulled high with a 10K Ohm resistor.

The Testmode pin should be pulled low. Typically a 470 Ohm resistor would be used.

## 8.2 Secondary device IDSel wiring

Each secondary PCI device has an IDSel pin which must be wired up to allow Configuration space accesses to occur. On Gamma the IDSel lines on multiplexed onto the SCIAD bus. The IDSels should be wired as follows:

- SCIAD28 should be connected to Idsel of a device fitted as secondary device 1.
- SCIAD29 should be connected to Idsel of a device fitted as secondary device 2.
- SCIAD30 should be connected to Idsel of a device fitted as secondary device 3.

## 8.3 GLINT side-band signals

GLINT devices have 2 sets of side-band signals for use after a Gamma device. These control data throttling into the input FIFO, and out of the output FIFO.

- The Gamma GLINTInDis0 should be connected to PCIFIFOInDis of the GLINT rendering device fitted as secondary device 1.
- The Gamma GLINTOutDis0 should be connected to PCIFIFOOutDis of the GLINT rendering device fitted as secondary device 1.
- The Gamma GLINTInDis1 should be connected to PCIFIFOInDis of a GLINT rendering device fitted as secondary device 2.
- The Gamma GLINTOutDis1 should be connected to PCIFIFOOutDis of the GLINT rendering device fitted as secondary device 2.

## 8.4 Using a secondary bus master

The Gamma device allows for one device to be a PCI bus master. To use PCI bus mastering the device must be connected to the second set of secondary bus PCI control signals. i.e. SCIFrameN(1), SCIIRdyN(1), SCITRdyN(1), SCIStopN(1), and SCIDevselN(1).

# 9 Thermal Characteristics

The maximum junction temperature must be kept below $T_j$(max) and this can only be guaranteed by proper analysis of the operating environment and the thermal path between the die and the air surrounding it.

## 9.1 Device Characteristics

These are fixed characteristics of the device and are independent of the operating environment or the characteristics of any heatsink:-

$T_{j(max)}$ = 100 °C.
Pd(max) = 6.5 Watts @ Vdd(max), $f_{MClk}$ = 125MHz, $f_{IOClk}$ = 66MHz.
$\theta_{jt}$ = 1.1 °C/Watt.

## 9.2 Thermal Model

The formula used to calculate the junction temperature ($T_j$) is

$$T_j = T_a + Pd( \theta_{jt} + \theta_{cs} + \theta_{sa} )$$
$$= T_a + Pd\theta_{ja}$$

Where:

$T_j$ = Junction temperature (°C)
$T_a$ = Ambient temperature (°C)
Pd = Power dissipation (Watts)
$\theta_{jt}$ = Junction to top of case thermal resistance (°C/Watt)
$\theta_{cs}$ = Case to Heatsink thermal resistance (°C/Watt)
$\theta_{sa}$ = Heatsink to Air thermal resistance (°C/Watt)
$\theta_{ja}$ = Total Junction to Air thermal resistance (°C/Watt)

The $\theta_{ja}$ form of the equation is more appropriate when there is no heatsink attached to the device.

## 9.3 Operation Without Heatsink

Gamma should not be operated without a heatsink attached.

## 9.4 Operation With Heatsink

With a heatsink attached to the device the junction temperature will depend on $\theta_{cs}$ and $\theta_{sa}$. $\theta_{cs}$ is the thermal resistance of the join between the heatsink and the case. $\theta_{sa}$ is the thermal resistance of the heatsink and will be a function of system airflow.

The table below shows the calculated thermal figures for a 27mmx27mm heatsink with 11.4mm high fins.

| Airflow lfpm | | $\theta_{ja}$ ° C/W |
|---|---|---|
| 0 | (Convection Cooling) | 9.2 |
| 50 | (0.25m/sec) | 8.1 |
| 100 | (0.5m/sec)` | 6.7 |
| 200 | (1.0m/sec)` | Not calculated |
| 400 | (2.0m/sec)` | Not calculated |

**Table 9-1 Operation With Heatsink**

This heatsink would be suitable for Gamma operation for ambient temperatures of up to 35$^{o}$C with zero airflow.

### 9.4.1 Heatsink Attachment

The following method has been approved for the purpose of attaching a heatsink directly onto the copper surface of the SBGA package:

Thermally conductive epoxy using either Loctite Output 315 with Loctite 7386 or type EG 7655 from A.I. Technology Inc. The thickness of the epoxy layer should be between 0.05mm and 0.15mm with 95% coverage of the contact area.

Typical achievable $\theta_{cs}$ using this method is 1.0 ° C/Watt

# 10 Index