Galileo Technology™

Switched Fast Ethernet Controller

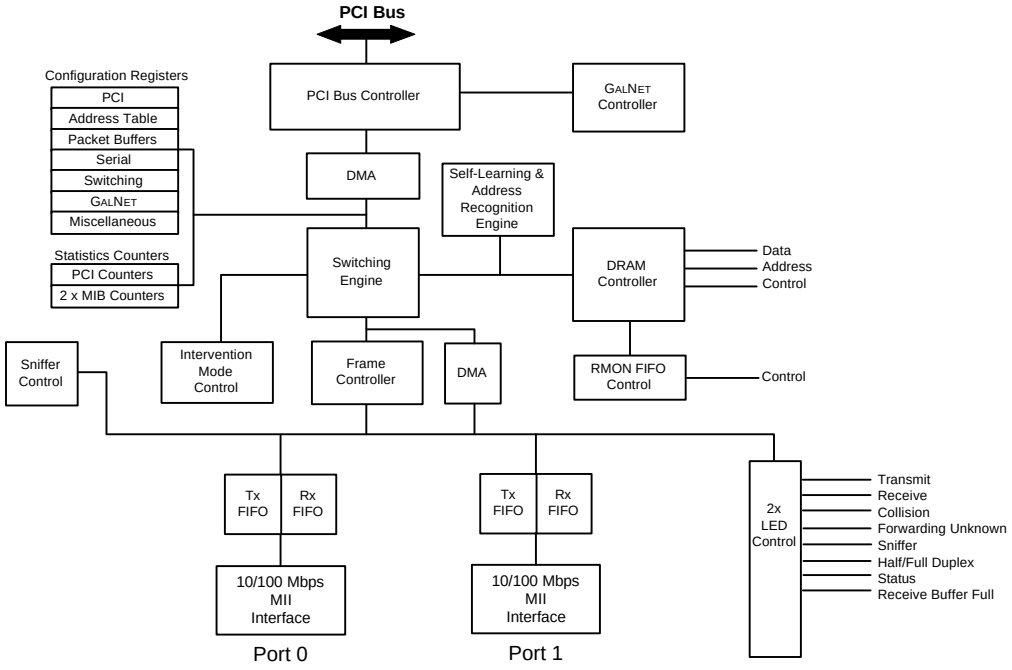
GT- 48002

Preliminary Rev. 1.1
October 2, 1996

NOTE: Always contact Galileo Technology for possible updates before starting a design.

FEATURES

- Member of GalNet™ Architecture Family
- Ideal uplink/server connect for the 10 Mbps GT-48001 Switched Ethernet Controller
- High integration, low cost, Switched Fast Ethernet Controller for 100 Base-TX, Base-T4, and Base-FX
- Provides switching functions between two 10/100Mbps, auto-negotiated Ethernet ports and PCI Bus
- Switch expansion via, high bandwidth PCI interface (up to 1Gbps)
- Supports 'Store and Forward' switching approach
- Low last-bit to first-bit delay
- Permits forwarding and filtering at full wire speed of 148,800 packets/sec on each Ethernet port
- Advanced address recognition
  - Intelligent address recognition mechanism enables forwarding rate at full wire speed
  - Self-learning mechanism
  - Supports up to 8K Unicast addresses and unlimited Multicast/Broadcast addresses
  - Broadcast storm rate filtering
- Permits software or hardware intervention in the packet routing decision mechanisms
- Incorporates two 802.3 compliant 10/100Mbps Media Access Controllers
  - Direct Interface to MII (Media Independent Interface)
  - Half/Full Duplex Support (up to 200 Mbps/port)
  - IEEE 802.3 100Base-TX, T4, and FX compatible
- Full MII Management Support (MDC/MDIO) via CPU access
- Auto-negotiation supported through MII Interface
- VLAN tagging support (1522 bytes)
- High observability LED interface
- 6 parallel LED outputs per port, including internal "monostable" function to enable viewing of dynamic signals
- 3 pin serial LED interface for additional status information per port.
- Direct support for packet buffering
  - Interfaces directly to 1Mbyte or 2Mbyte 32-bit, 60ns EDO DRAM
  - Up to 1K buffers, 1536-bytes each, dynamically allocated to the receive and PCI ports
- Interfaces directly with PCI Rev 2.1 for switch expansion
  - Up to 4 GT-48001/ GT-48002 devices per PCI slot without PCI-to-PCI bridging, and up to 32 GT-48001/ GT-48002 devices in a switch
  - CPU connection for management
  - Connection to other media
- Various management support features
  - Repeater MIB and PCI counters, plus enhanced counter set
  - Aging support
  - Hardware assist for Spanning Tree algorithm(IEEE 802.1)
  - Station-to-Station connectivity matrix (for RMON)
  - CPU access to Address Table
  - Ability to define static addresses
  - Monitoring (Sniffer) mode
- 33MHz clock rate
- 5V operation
- 208 pin QFP package



OVERVIEW

General Description

The GT-48002 is a high performance, low cost, Switched Fast Ethernet Controller. It provides the switching functions between two dedicated 10/100Mbps Ethernet ports. Switch expansion to 10Mbps Ethernet ports (using the GT-48001) or to 10/100Mbps ports (using additional GT-48002's) is enabled via a high performance, high bandwidth PCI bus.

The GT-48002 is a member of Galileo's GalNet™ switching architecture. GalNet™ enables designers to build intelligent and scalable switching hubs and lets the user boost performance cost-effectively in departments, workgroups and small data centers.

The GT-48002 uses a Store and Forward switching approach. It forwards and filters faster than than full wire speed.

Serial Interface

The GT-48002 incorporates two 10/100Mbps Ethernet ports. Two Media Independent Interfaces (MII) are provided for glueless connection to off-the-shelf PHY chips. Each port includes the Media Access Control function (MAC) and six LEDs for Link Status, Collision, Receive Transmit, Half/Full Duplex and Receive Buffer Full indications.

The GT-48002 incorporates full MII management support. The MDC/MDIO pins are directly controlled by the CPU.

Address Recognition

The GT-48002 supports up to 8K different MAC addresses and unlimited Multicast/Broadcast addresses. An intelligent address recognition mechanism enables filtering and forwarding at full wire speed.

The GT-48002 provides a self address learning mechanism. Each device holds its own Address Table. The GT-48002 learns the new addresses as they arrive from the wire and updates all the Address Tables in the system.

DRAM Interface

The GT-48002 interfaces directly to a 1Mbyte or 2 Mbyte DRAM. The DRAM is used to store the incoming/outgoing packets as well as the Address Table. The on-chip DRAM controller supports EDO DRAMs.

The GT-48002 supports up to 1008 (full packet size) receive buffers. These buffers are dynamically allocated to the receive ports and the PCI port. The number of buffers can be optionally limited on a per port basis.

PCI Interface

The GT-48002 has a glueless connection to the PCI bus. The interface is compliant with PCI Rev 2.1. The GT-48002 can be either a master initiating a PCI bus operation or a target responding to a PCI bus operation. The PCI bus is used as a backplane to expand the switch allowing connection to 10Mbps Ethernet ports (using the GT-48001) or 10/100Mbps ports. Up to four GT-48001/ GT-48002 devices can reside on the same PCI bus, forwarding packets from one port to the other. By using PCI-to-PCI bridge devices, the switch can be expanded to up to 32 devices

The PCI bus may also be used to connect a CPU for management / routing functions, and to connect to other LAN technologies such as ATM or FDDI. The connection to a CPU is optional.

Management Features

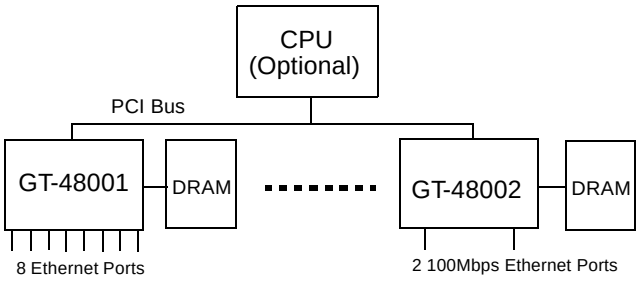
The GT-48002 provides comprehensive management capabilities enabling the user to manage and balance the network.

The GT-48002 supports per-port statistics counters and PCI traffic counters. It implements hardware assistance for Aging, Spanning Tree algorithms and allows the CPU access to the Address Table. The GT-48002 also provides a Station-to-Station Connectivity matrix (RMON) and the ability to select a port to work in Monitoring (Sniffer) mode.

Intervention Mode

The GT-48002 incorporates an enhanced feature called 'Intervention' mode. This feature permits software or hardware intervention in the packet routing decision. Intervention is performed differently for Multicast and for Unicast packets. Multicast packets are forwarded *only* to the CPU. The CPU forwards the packets to selected ports in the GT-48002 devices.

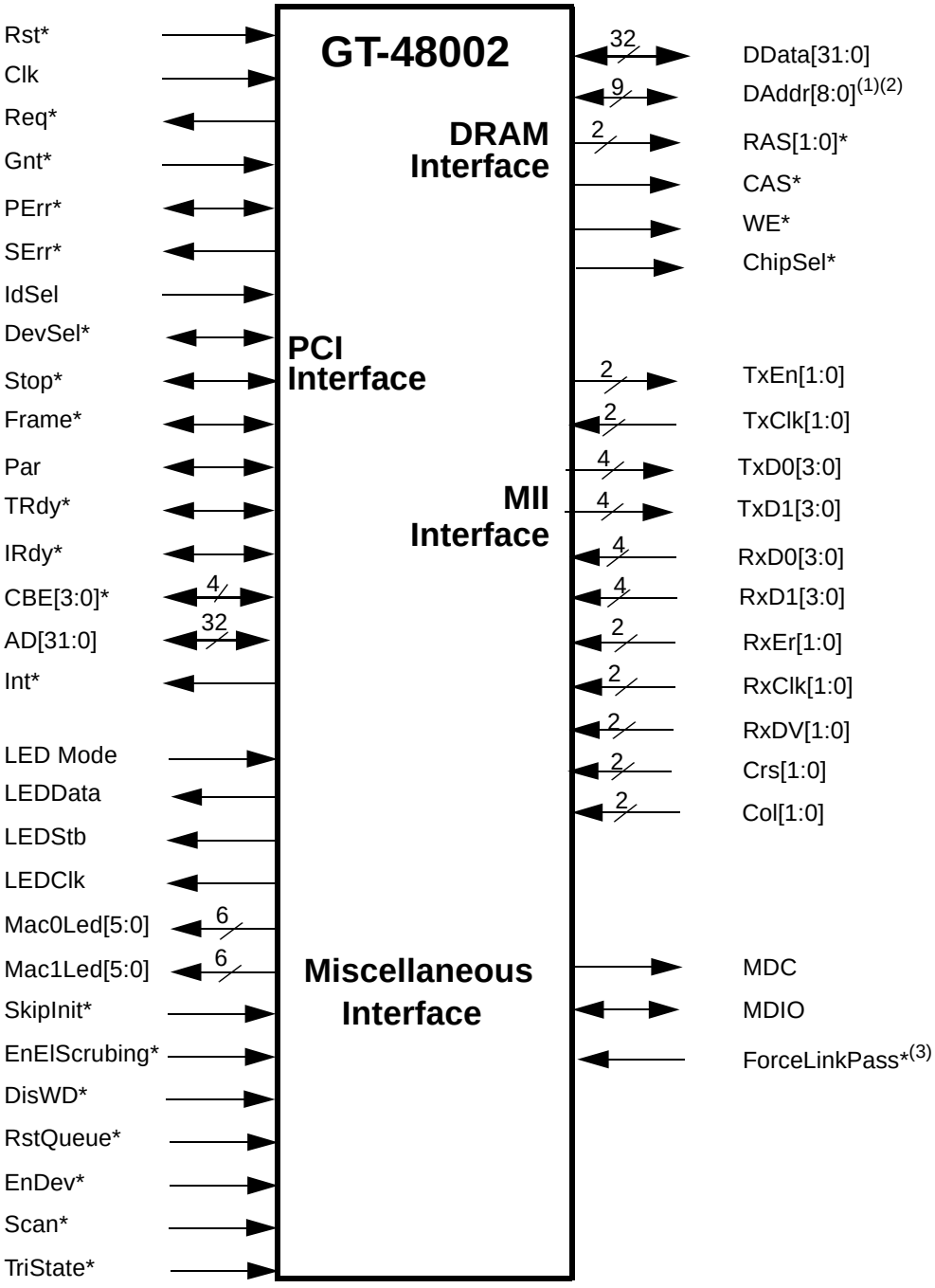
For Unicast packets, the CPU has the ability to modify the routing decision. It can discard the packet, it can change the target port, or it can request the entire packet information.



Easy Expansion To 256-Ports and CPU Connectivity Via the PCI Interface

1 PIN INFORMATION

1.1 Logic Symbol



(1) - Pins[8,5:0] sampled at reset to establish Device Number and DRAM Parameters
(2) - Pins[7:6] sampled at reset to establish Half/Full Duplex Mode per port (1= full duplex)
(3) - Pin sampled at reset to force the link status of all ports to link up state (0 = force link up)

1.2 Pin Assignment Table

Symbol	Type	Description
PCI Interface		
Rst*	I	**Reset:** Active low. Resets the GT-48002 to its initial state. Rst* must be asserted for at least 10 PCI clock cycles. When in the reset state, all PCI output pins are tristated and all open drain signals are floated. Following Rst* deassertion, the GT-48002 clears the internal buffers and initializes the address table in the DRAM. The address table initialization takes 165,000 CLK cycles to complete. Any incoming packets during the address table initialization, are ignored.
Clk	I	**Clock:** Provides the timing for the GT-48002 internal units. All units except for the Serial Interfaces use this clock. It also provides the timing for the PCI-related bus transactions. The clock frequency is 33MHz
Req*	O	**Bus Request:** Asserted to indicate to the PCI bus arbiter that the GT-48002 desires use of the bus.
Gnt*	I	**Bus Grant:** Indicates to the GT-48002 that access to the PCI bus is granted.
PErr*	I/O	**Parity Error:** Asserted when a data parity error is detected on the PCI bus.
SErr*	O	**System Error:** Asserted when an address parity error is detected. The GT-48002 asserts the SErr* two cycles after the failing address. This output has an open-drain output driver.
IDSel	I	**Initialization Device Select:** Acts as a chip select during PCI configuration read and write transactions.
DevSel*	I/O	**Device Select:** Asserted by the target of the current access. When the GT-48002 is a bus master, it expects the target to assert DevSel* within 5 bus cycles, confirming the access. If the target does not assert DevSel* within the required bus cycles, the GT-48002 aborts the cycle. As a target, when the GT-48002 recognizes its transaction, it asserts DevSel* in a “medium speed” (two cycles after the assertion of Frame*).
Stop*	I/O	**Stop:** Indicates that the current target is requesting the bus master to stop the current transaction. As a master, the GT-48002 responds to the assertion of Stop*, either disconnecting, retrying, or aborting. As a target, the GT-48002 asserts Stop* to retry.
Frame*	I/O	**Cycle Frame:** Asserted by the GT-48002 to indicate the beginning and duration of a master transaction. Frame* is asserted to indicate the beginning of the cycle. While Frame* is asserted, data transfer continues. Frame* is deasserted to indicate that the next data phase is the final data phase transaction. Frame* is monitored when the GT-48002 acts as a target, to detect a configuration or memory transaction.
Par	I/O	**Parity:** Calculated by the GT-48002 as an even parity bit for the AD[31:0] and CBE[3:0]* lines.

Symbol	Type	Description
TRdy*	I/O	**Target Ready:** Indicates the target agent's ability to complete the current data phase of the transaction. A data phase is completed on any clock when both TRdy* and IRdy* are asserted. Wait cycles are inserted until both IRdy* and TRdy* are asserted together.
IRdy*	I/O	**Initiator Ready:** Indicates the bus master's ability to complete the current data phase of the transaction. A data phase is completed on any clock when both TRdy* and IRdy* are asserted. Wait cycles are inserted until both IRdy* and TRdy* are asserted together.
AD[31:0]	I/O	**Address/Data:** 32-bit multiplexed PCI address and data lines. During the first clock of the transaction, AD[31:0] contains a physical byte address (32 bits). During subsequent clock cycles, AD[31:0] contains data.
CBE[3:0]*	I/O	**Bus Command/Byte Enable:** These signals are multiplexed on the same PCI pins. During the address phase of the transaction, CBE[3:0]* provide the Bus Command. During the data phase, they provide the Byte Enables, which determine which bytes carry valid data.
Int*	O	**Interrupt Request Line:** Int* is asserted by the GT-48002 when one (or more) of the bits in the Interrupt Cause register is set. This output has an open-drain output driver.
DRAM Interface		
DData[31:0]	I/O	**DRAM Data:** 32-bit EDO DRAM Data.
DAddr[8:0]	I/O	**DRAM Address Bus:** In normal operation, DAddr[8:0] contain the DRAM address. During reset, these multiplexed pins are sampled by the GT-48002 to indicate the Device Number, DRAM Parameters, and the Duplex Mode (see Reset Configuration section). Values are determined by connecting the appropriate pull-up/pull-down resistors. The configuration information is accessible via the Status and Port Control registers.
RAS[1:0]*	O	**Row Address Strobes:** Active low. DRAM row address strobes. RAS[0]* is used for Bank0. RAS[1]* is used for Bank1.
CAS*	O	**Column Address Strobes:** Active low. DRAM column address strobe.
WE*	O	**Write Enable:** Active low. DRAM write enable.
ChipSel*	O	**Chip Select:** Active low. This pin is connected to an external FIFO device for buffering data for RMON. It is asserted by the GT-48002 to indicate that the packet's Byte Count, Destination Address and Source Address are being read from the DRAM. This information is stored in the FIFO and accessed by an external CPU for Station-to-Station connectivity matrix implementation.
Media Independent Interface		
TxEn[1:0]	O	**Transmit Enable:** Active high. This output indicates that the packet is being transmitted. TxEn is synchronous to TxClk.
TxClk[1:0]	I	**Transmit Clock:** Provides the timing reference for the transfer of TxEn, TxD signals. TxClk frequency is one fouth of the data rate (25 MHz for 100Mbps, 2.5 MHz for 10Mbps). TxClk nominal frequency should match the nominal frequency of RxClk for the same port.

Symbol	Type	Description
TxD0[3:0]	O	**Transmit Data 0:** Outputs the Port0 Transmit Data. Synchronous to TxClk[0].
TxD1[3:0]	O	**Transmit Data 1:** Outputs the Port1 Transmit Data. Synchronous to TxClk[1].
Col[1:0]	I	**Collision detect:** Active high. Indicates a collision has been detected on the wire. This input is ignored in full-duplex mode.
RxD0[3:0]	I	**Receive Data 0:** Port 0 Receive Data. Synchronous to RxClk[0].
RxD1[3:0]	I	**Receive Data 1:** Port 1 Receive Data. Synchronous to RxClk[1].
RxEr[1:0]	I	**Receive Error**. Active high. Indicates that an error was detected in the received frame. This input is ignored when RxDV for the same port is inactive.
RxClK[1:0]	I	**Receive Clock**. Provides the timing reference for the transfer of the RxDV,RxD,RxEr signals (per port). Operates at either 25 MHz (100Mbps) or 2.5 MHz (10Mbps). The nominal frequency of RxClk (per port) should match the nominal frequency of that port's TxClk.
RxDV[1:0]	I	**Receive Data Valid:** Active high. Indicates that valid data is present on the RxD lines. Synchronous to RxClk.
CrS[1:0]	I	**Carrier Sense:** Active high. Indicates that either the transmit or receive medium is non-idle.
MDC	O	**Management Data Clock:** Provides the timing reference for the transfer of the MDIO signal. This output may be connected to the PHY devices of both ports.
MDIO	I/O	**Management Data Input/Output:** This bidirectional line is used to transfer control information and status between the PHY and the GT-48002. It conforms with IEEE Std 802.3. This signal may be connected to the PHY devices of both ports. When not in use, this pin must be connected to a pull-down resistor.
Misc. Interface		
LEDMode	I	**LED Mode Select:** Affects Port status LED, LEDClk frequency and LED ON time values. 0 - select LED mode 0 1 - select LED mode 1
LEDData	O	**LED Data:** Active low. Serial data bit stream which contains the LED indicators per port. The data is shifted out using the LEDClk. LEDStb is used to mark the first data bit.
LEDStb	O	**LED Strobe:** Active high. Indicates the beginning (data bit #1) of a valid data frame on LEDData output.
LEDClk	O	**LED Clock:** 1 MHz clock (at LED mode 0), 202 KHz clock (at LED mode 1). This output is used to clock the LEDStb and LEDData outputs. During RESET, LEDClk frequency is 33 MHz.
Mac0LED[5:0]	O	**MAC LED 0:** Active LOW LED outputs for port 0 bit [0] : Port Status (operation according to LEDMode) bit [1] : Transmit In Progress (TxEn active) bit [2] : Receive In Progress (RxDV active) bit [3] : Collision (Col active) bit [4] : Full Duplex (port configured to Full Duplex) bit [5] : Receive Buffer Full (programmable limit exceeded) An external driver is required to drive the LEDs.

Symbol	Type	Description
MAC1LED[5:0]	O	**MAC LED 1:** Active LOW LED outputs for port 1 Same as MAC0LED
RstQueue*	I	**Reset Transmit Queues:** When asserted, all internal transmit and receive queues are cleared. All GT-48002 state machines are reset to their initial state. This function may be also controlled by software via the Global Control Register.
ForceLinkPass*	I/O	**Force Link Pass:** Active low. This pin is sampled at reset. When connected to a pullup, the link status of the ports is read through the SMI (MDC/MDIO interface) from the PHY devices (register#1, bit#2). When connected to a pull-down, the link status of all ports remains in the “link is up” state regardless of the PHY’s link bit value. This pin should be connected to either a pull-up (normally) or a pull-down resistor (to force the link pass). Following Rst* deassertion, this pin becomes an output (unused - value is undefined).
EnDev*	I	**Enable Device:** Active low. Enables serial and PCI ports. When asserted, all serial ports and the PCI port are active. When deasserted, both the ports and the PCI interface are disabled. This function may be also controlled by software via the Control Register.
DisWD*	I	**Disable Watchdog timer:** Active low. When asserted the Tx Watchdog timers operation is disabled.
EnELScrub*	I	**Enable Empty List Scrubing:** Active low. When asserted, the empty list scrubbing mechanism is enabled. For testing purposes only. Must be pulled HIGH.
SkipInit*	I	**Skip Initialization:** Active low. When asserted, the GT-48002 skips the Address Table initialization sequence. This pin is used only for testing and should be driven high for normal operation.
Scan*	I	**Scan:** This pin together with TriState* indicates the GT-48002 mode of operation as follows: Scan* = 1, TriState* = 1 - Normal operation Scan* = 0, TriState* = 1 - Reserved. Scan* = 1, TriState* = 0 - The GT-48002 drives all outputs and I/O pins to High impedance. Scan* = 0, TriState* = 0 - Reserved.
TriState*	I	**Tri State:** This pin together with Scan* indicates the GT-48002 mode of operation as described above.

2 FUNCTIONAL DESCRIPTION

2.1 General Description

The GT-48002 provides the switching functions between two 10/100 Mbps Ethernet ports. The GT-48002 has three interfaces:

a) Two 10/100 Mbps Ethernet ports that interface directly to the MII ports.

b) 32-bit DRAM to store the Address Table and the incoming/outgoing packets.

c) PCI bus for switch expansion, connectivity, and management.

The GT-48002 receives the incoming packets from the MII, searches in the Address Table the Destination Address and forwards the packet to the appropriate port. The port can be either local or in different GT-48001/2 devices that reside on the same PCI bus. In parallel, the GT-48002 checks the Source Address and updates all the Address Tables in the system accordingly.

The GT-48002 uses a simple protocol on the PCI, consisting of 5 messages. This protocol is an essential part of the GalNet™ architecture. The 5 messages are: 'new_address', 'buffer_request', 'start_of_packet', 'packet_transfer', and 'end_of_packet'.

2.2 Address Table

The GT-48002 supports up to 8K different MAC addresses. The Address Table is located in the DRAM and is fully controlled by the GT-48002 (i.e. a new address is automatically added to the Address Table). The CPU has the ability to insert, remove or modify the entries. Figure 1 shows the Address Table structure.

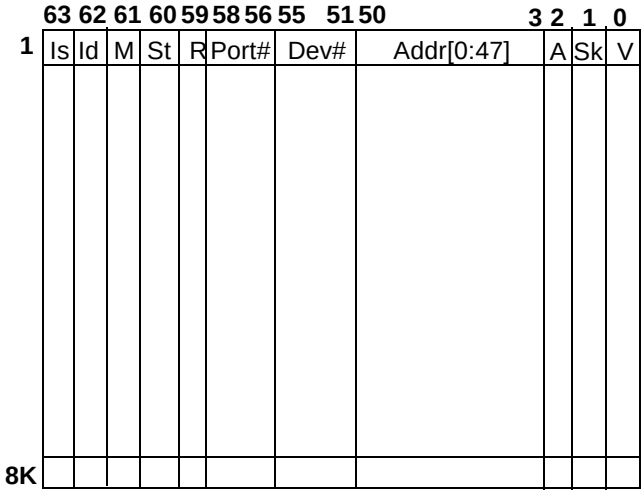


Figure 1. Address Table Structure

Bit	Description
V	Valid - Indicates that the entry is valid 0 - Not Valid 1 - Valid
Sk	Skip - Skip this entry, used to delete an entry 0 - Don't skip this entry 1 - Skip this entry
A	Aging - This bit is used for the Aging process. - Set by the GT-48002 upon receiving a packet from the station corresponding to this entry. - Cleared by the CPU
Addr	Address - 48 bits of MAC address.
Dev#	Device Number - Indicates which of a maximum of 32 devices in the system is associated with this address.
Port#	Port Number - Indicates which of the two ports in a GT-48002 is associated with this address.
R	Reserved
St	Static - Indicates whether an entry can be modified or not. 0 - The entry can be modified 1 - The entry is static. The Dev# and Port# cannot be modified
M	Multiple - Meaningful when bit St is set. 0 - Forward this packet only to the destination port 1 - Forward this packet to all ports (as Unknown)
Id	Intervention for Destination Addresses 0 - Don't activate the Intervention mode 1 - Activate the Intervention mode
Is	Intervention for Source Addresses 0 - Don't activate the Intervention mode 1 - Activate the Intervention mode

2.2.1 Learning Process

The GT-48002 has a self-learning mechanism. It learns the Ethernet addresses in real time. The GT-48002 searches for the Source Address (SA) in the Address Table and acts as follows:

1. If the SA was not found in the table (a new address), the GT-48002 waits to the end of the packet (good CRC) and updates its Address Table. It also notifies the other GT-48001/2 devices and the CPU by sending a

'new_address' message on the PCI interface. The message contains the MAC address, the Device Number and the Port Number (the message format is described in section 2.10). In addition, the GT-48002 asserts Int* to notify the CPU that the Address Table was modified.

2. If the SA was found, the GT-48002 compares the Port Number and the Device Number to the device and Port Numbers on which the packet was received. If they are different, and the St bit in the Address Table is cleared, it updates the entry with the new information and notifies the other GT-48001/2 devices and the CPU. If they are equal, no action is taken.

3. If the SA was found in the Address Table, the Aging bit is set.

The CPU can access the Address Table to modify, remove or to add a MAC address. This is done by performing a 'new_address' message on the PCI.

2.2.2 Address Recognition

The GT-48002 forwards the incoming packets to the appropriate ports(s) according to Destination Address (DA) as follows:

1. If the DA is a Unicast address and the address was found, the GT-48002 acts as follows:
   - If the Port Number and the Device Number are equal to the Port/Device on which the packet was received, the packet is discarded.
   - If the Port Number is different, but the Device Number is equal, the packet is forwarded to the appropriate local port.
   - If the Device Number is different, the packet is forwarded to the appropriate GT-48001/2 device via the PCI bus.

2. If the DA is a Unicast address and the address was not found (Unknown), the GT-48002 acts as if it is a Multicast packet. This packet is forwarded to the ports (except for the port in which the packet was received) and the devices which were programmed for forwarding of unknown packets (bit 7 in the Port Control registers).

3. If the DA is a Multicast address, the packet is forwarded to all the local ports (except for the port in which the packet was received). It is also forwarded to all other GalNet™ devices via the PCI bus.

2.2.3 Recovery Process

The purpose of the Recovery Process is to guarantee that Address Tables in all the devices will be similar. When the packet is Unknown, the source GT-48002 sends a 'new_address' message to all the devices. Each device searches its own table for the new address. More than one device can find the address, but only one device owns this address (i.e. the Device Number written in the Address Table for that address is equal to its own Device Number). This particular device updates the source GT-48002 Address Table with the new address (by sending it and only it a 'new_address' message).

2.3 GT-48002 Buffers and Queues

The GT-48002 incorporates three transmit queues (for the 2 local ports and the PCI), and one common receive buffer. Figure 2 shows the GT-48002 queues. The receive buffers as well as the transmit queues are located in the DRAM. The GT-48002 includes the pointers to the transmit queues. The GT-48002 data structure components are the following:

1. Receive Buffer - A common Rx buffer for all ports. The common Rx buffer is divided into 320 or 1008 separate buffers (depending on the DRAM size) of 1.5KBytes (1536 bytes) each. Each separate buffer contains the whole packet information.

2. Rx Empty List - A list of 320 or 1008 bits. Each bit contains the status of its appropriate receive buffer in the DRAM (empty or occupied).

3. Tx Descriptors - A set of 3 transmit descriptor rings. Each ring contains 1008 descriptors. The descriptor size is 1 Long Word (32-bits) and contains the Buffer Address divided by 600hex (1.5K), the Byte Count and the Packet Type (Multicast or Unicast).

4. Read/Write Pointers - 3 pairs of pointers to the transmit descriptors.

2.4 Packet Forwarding

The following chapter describes the procedures for forwarding packets between local ports, between GT-48001/2 devices, and between the CPU and a GT-48002.

2.4.1 Forwarding a Unicast Packet to a Local Port

The sequence is as follows:

1. The incoming packet is fed to the Receive FIFO (there is a 20x32-bit FIFO per port) and is transferred to an empty block in the DRAM in 8 32-bit bursts.

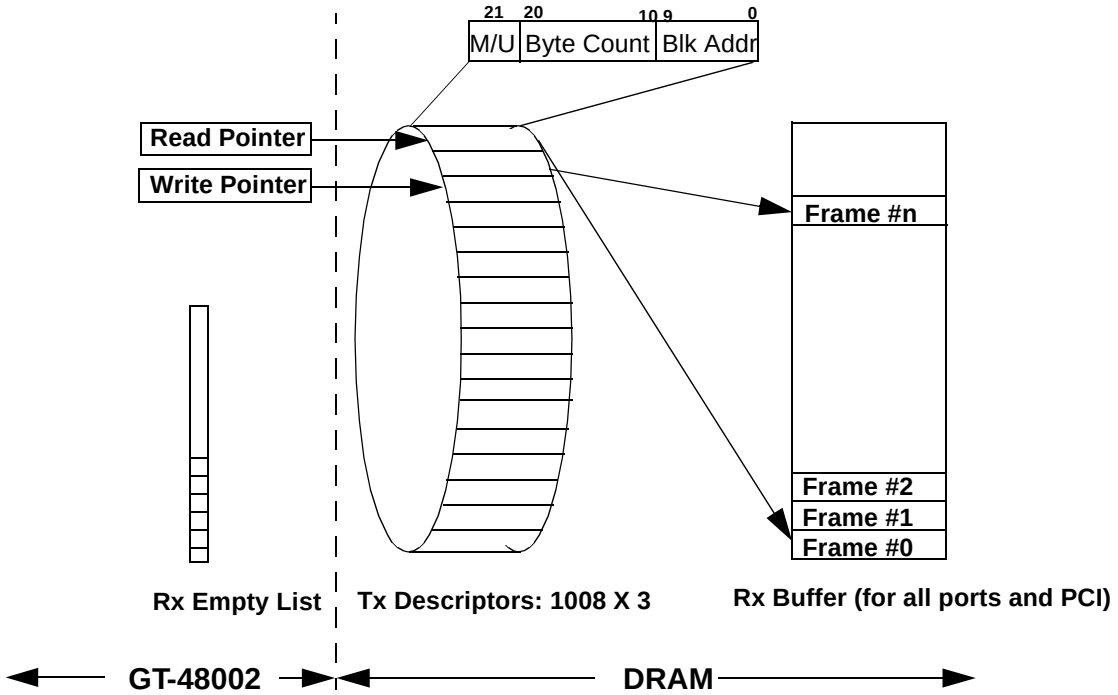


Figure 2 - GT-48002 Buffers and Queues

2. In parallel, an address recognition cycle is performed for both the DA and the SA. The GT-48002 uses the DA's Port Number and Device Number to queue the packet to the appropriate port.

3. At the end of a good packet transfer, the packet is forwarded to the appropriate transmit queue. This is done by writing the Byte Count and the Block Address to the Tx Descriptor which is pointed to by the Write Pointer.

4. The Write Pointer is incremented. The GT-48002 transmits when the Write Pointer is not equal to the Read Pointer.

5. At the end of the packet transmit process, the GT-48002 increments the Read Pointer and clears the appropriate bit in the Empty List.

2.4.2 Forwarding a Unicast Packet to a Port in a Different Device

The sequence is as follows:

1. The incoming packet is fed to the Receive FIFO and transferred to an empty block in the DRAM in 8 32-bit bursts.

2. In parallel, an address recognition cycle is performed for both the DA and the SA. The GT-48002 uses the DA's Port Number and Device Number to queue the packet to the appropriate GT-48002 device and port.

3. At the end of a good packet transfer, the packet is entered into the PCI transmit queue (the third queue). This is done by writing the packet information to the PCI transfer request descriptor which is pointed to by the Write Pointer. When the Write Pointer is not equal to the Read Pointer, the source device sends a 'buffer_request' message to the appropriate target device indicating that there is a packet for transmission.

4. The target device allocates a buffer in its DRAM and sends a 'start_of_packet' message to the source device.

5. The source device transfers the packet using PCI master operations in 8 32-bit bursts. At the end of the packet, the source device performs an additional write transaction ('end_of_packet' message) and places on the PCI the Byte Count, the target Port Number, the Receive Block address, and the Packet Type. It also frees its packet buffer by clearing the appropriate bit in its Empty List.

6. The packet is entered to the appropriate transmit queue in the target device. This is done by writing the Byte Count and the Rx Buffer address to the Tx Descriptor which is pointed to by the Write Pointer.

7. The Write Pointer is incremented. The target GT-48002 transmits when the Write Pointer is not equal to the Read Pointer.

8. At the end of the packet transmit process, the GT-48002 increments the Read Pointer and frees its packet buffer by clearing the appropriate bit in its Empty List**.**

2.4.3 Forwarding a Multicast Packet

The GT-48002 forwards the Multicast packets to all the ports and devices using the same mechanism as in Unicast packets. The packet is queued to all local transmit ports except for the port in which the packet arrived. All the GT-48001/2 devices in the system receive a 'buffer_request' message, allocate a buffer in their DRAM and send back a 'start_of_packet' message. The packet is transferred separately to each GT-48001/2 device in the system.

2.4.4 Forwarding a Packet to the CPU

The GT-48002 forwards the packets directly to the CPU main memory. The GT-48002 contains two pointers to a sixteen block buffer area in the memory (Shadow and Base Address). Figure 3 shows the data structure in the CPU main memory.

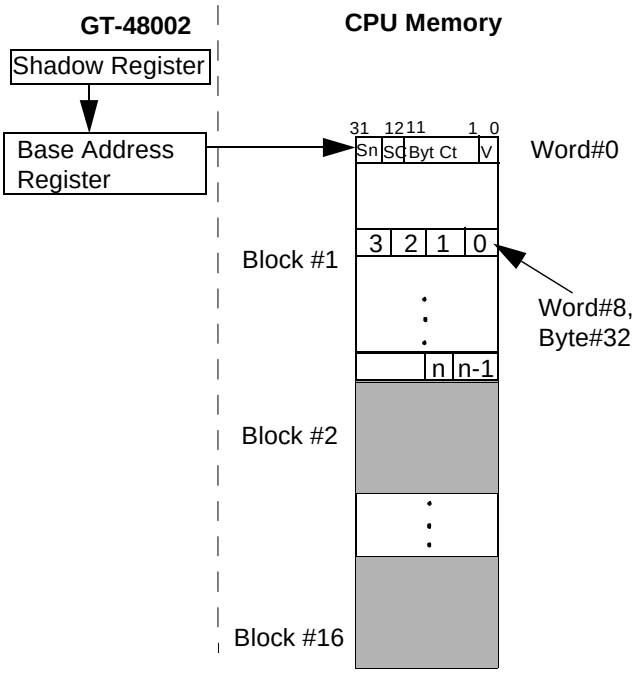


Figure 3 - CPU Data Structure

The data structure components are the following:

1. Base Address Register - A register that points to the beginning of a sixteen block area in the CPU main memory.

2. Shadow Register - A second register that holds a pointer to a second sixteen block area in the CPU main memory. The value in the Shadow register propagates into the Base Address register after sixteen packets are transferred to the main memory.

3. Buffer Area - It consists of 16 blocks of 2Kbytes each. The first word in the block contains the Byte Count (bits [11:1]), Source Channel (bit 12), Sniffer bit (bit 13) and a Valid bit (bit 0). The Source Channel, Byte Count and the Valid bit are written at the end of the packet transfer. Bit [31] is the Sniffer bit and will be set to '1' to mark a sniffer packet when the CPU is configured as the target sniffer. Words 1 to 7 are left empty for user purposes.The packet data is written starting the first byte in the 8th word (32nd byte).

The communication between the GT-48002 and the CPU follows this sequence:

1. CPU updates the Base Address register

2. CPU updates the Shadow Register. The Shadow register is loaded when the CPU writes to the same address of the Base Address register, after it was written into. (Reading the value from the Base Address register address will result with the last value written).

3. GT-48002 transfers 16 packets to the CPU main memory and asserts the Int* at the end of each packet transfer.

4. The CPU counts sixteen interrupts and updates the Shadow register.

Steps 3-4 are repeated. The packet transfer to the CPU is done as follows:

1. The incoming packet is fed to the Receive FIFO and is transferred to an empty block in the DRAM in 4 32-bit bursts.

2. In parallel, an address recognition cycle is performed for both the DA and the SA. The GT-48002 uses the DA's Port Number and Device Number to queue the packet to the appropriate device and port.

3. At the end of a good packet transfer, the packet is entered into the PCI transmit queue. This is done by writing the packet information to the PCI transfer request descriptor which is pointed to by the Write Pointer. When the Write Pointer is not equal to the Read Pointer, the source device transfers the packet to the appropriate block in the CPU main memory. The data is entered into the 8th word (32nd byte). Words 1 to 7 are left empty for user purposes.

4. At the end of the packet transfer, the GT-48002 writes the Byte Count and the Valid bit to the first word of the block. It also sends an interrupt via Int* to the CPU and increments the Read Pointer and clears the appropriate bit in its Empty List.

2.4.5 Receiving a Packet from the CPU

The CPU forwards a packet to the ports using the same mechanism as the other GT-48002 devices with one exception. The 'start_of_packet' message from the target device to the CPU is transferred directly to the CPU main memory. The GT-48002 holds a Start of Packet Base Address register which points to a buffer area in the CPU main memory. The buffer area can hold up to 32 'start_of_packet' messages.

2.5 Tx Watchdog Timer

The GT-48002 holds a Transmit Watchdog timer for each transmit queue. For 100 Mbps operation, the default value of the timer is 63msec and the range is between 10.5mSec to 168msec, in 10.5 mSec steps. For 10 Mbps operation, the default value of the timer is 630msec and the range is between 105mSec to 1680msec, in 105 mSec steps. The timer measures the time between two consecutive packets which are being served. When the timer expires, the GT-48002 clears the appropriate used blocks and sends an interrupt to the CPU by asserting Int*.

2.6 GT-48002 Device Table

The GT-48002 includes a 32-bit Device Table. Each bit in the table represents a different GT-48001/2 device in the system. Upon reset, each GT-48001/2 sets all the bits to '1'. The bits are cleared either by the CPU or upon PCI master abort (meaning that the GT-48002 tried to access a non-existent device). The Device Table is used by the GT-48002 to know whether or not to transfer a multicast address or new address to the target device. Bit 0 in the Device Table register corresponds to Device#0, etc.

2.7 Intervention Mode

The GT-48002 supports a powerful mode named Intervention Mode, which permits software or hardware intervention in the packet routing decision mechanisms. Intervention mode handles Multicast and Unicast packets differently as shown below.

2.7.1 Multicast Packets

When the routing Intervention option is set for Multicast packets, all Multicast packets will be forwarded to the CPU memory. The CPU can decide to what ports the packets need to be sent. Only one packet needs to be sent to each GT-48002 device and each GT-48002 will automatically forward the packet only to the ports that the CPU tagged for that specific Multicast packet. These ports are tagged in bits [29:22] at the 'end_of_packet' message. The GT-48002 will enter the routing Intervention mode when bit 22 in the Global Control register is set. Figure 4 illustrates a Multicast packet transfer in routing Intervention mode.

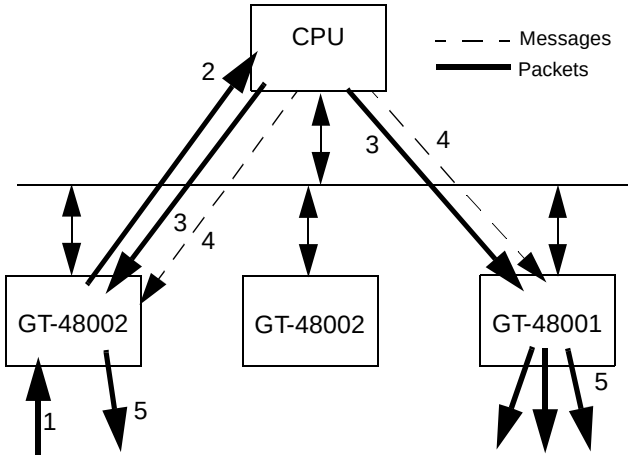


Figure 4 - Multicast Packet Transfer With Intervention

The sequence illustrated is as follows:

a. The Multicast packet is received and stored in the source GT-48002's DRAM (arrow #1).

b. The GT-48002 transfers the packet to the CPU main memory (arrow #2).

c. The CPU transfers the packet to the selected GT-48001/2 devices (arrows #3).

d. At the end of the packet transfer, the CPU sends an 'end_of_packet' message to tag the selected ports in which the packets will be transmitted (arrows #4).

e. The packet is transmitted on the selected ports (arrows #5).

2.7.2 Unicast Packets

Intervention in Unicast traffic is optional per MAC address (either Source or Destination Address). The entry in the Address Table includes two Intervention bits, one for the Source Address (bit 63) and one for the Destination

Address (bit 62). The Intervention mode can be activated based on Source Addresses, Destination Addresses or both. When one of the Intervention bits is set, the GT-48002 will not forward the packet automatically to the destination device. Instead, it will send a 'buffer_request' message to the CPU memory. The 'buffer_request' includes information about the routing of the packet (Source and Target port/device numbers). The CPU can have the following options: it can discard the packet, it can send a 'buffer_request' to the destination GT-48001/2 or any other device to take the packet, or it can request the entire packet information (data and headers) and modify it.

The 'buffer_request' messages will be sent to the buffer area in the CPU main memory which contains 256 entries of dual 32-bit words. The buffer area is pointed to by a Base Address register and a Shadow register.

Figure 6 shows a Unicast packet transfer in the Intervention mode.

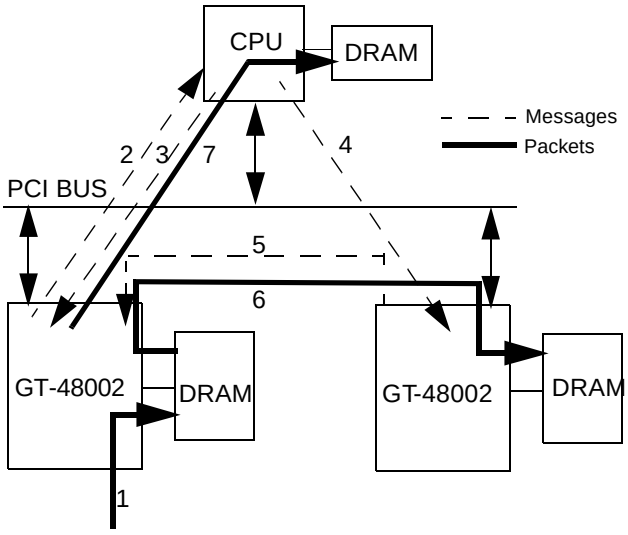


Figure 5 - Unicast Packet Transfer With Intervention

The sequence illustrated is as follows:

a. The Unicast packet is received and stored in the source GT-48002's memory (arrow #1).

b. If one of the Intervention bits is set, the GT-48002 sends a 'buffer_request' to the CPU (arrow #2). The 'buffer_request' includes the source port and the destination port/device.

The CPU then has the following options:

- Discard the packet (arrow #3). The CPU sends a 'start_of_packet' message with the Byte Count field cleared.

- Forward the packet to a destination device (arrow #4). The CPU sends a 'buffer_request' to the destination device. The destination device allocates a buffer and sends a 'start_of_packet' message to the source GT-48002 (arrow #5). The source GT-48002 device transfers the packet followed by an 'end_of_packet' message (arrow #6).

- Take the packet. The CPU sends a 'start of packet' message**.** The target device in the message is the CPU number. The source GT-48002 device sends the packet followed by an 'end_of_packet' message directly to the CPU main memory (arrow #7).

2.8 Management Support

The GT-48002 supports the following management features:

- MIB and PCI counters
- Station-to-Station connectivity matrix
- Monitoring (Sniffer)
- Spanning Tree

2.8.1 Repeater MIB and PCI Counters

The GT-48002 incorporates a full set of MIB counters per port, as well as PCI counters. The PCI counters are global for the two ports. The CPU periodically reads the counters which are then cleared automatically by the GT-48002 following the READ operation. The Repeater MIB counters are the following:

- Bytes Received
- Bytes Sent
- Frames Received
- Frames Sent
- Total Bytes Received (Good and Bad)
- Total Frames Received (Good and Bad)
- Multicast Frames Received
- Broadcast Frames Received
- CRC + Alignment Error
- Oversize Frames
- Fragments
- Jabber Frames
- Collision
- Late Collision
- Frames with length of 64 Bytes
- Frames with length of between 65-127 Bytes
- Frames with length of between 128-255 Bytes
- Frames with length of between 256-511 Bytes
- Frames with length of between 512-1023 Bytes

- Frames with length of between 1024-1522 Bytes
- MAC Rx Error (received packets with RxEr asserted)
- Dropped Frames

The global PCI counters are:

- PCI Frames Received
- PCI Frames Sent

2.8.2 Station-to-Station Connectivity Matrix

The GT-48002 provides a mechanism to record the Destination Address, Source Address and the Byte Count of all the forwarding packets in an external FIFO for RMON support. The FIFO is connected to the DRAM's data lines and controlled directly by the GT-48002. The GT-48002 asserts the ChipSel* pin, and reads in two consecutive transactions the Destination Address and Source Address (a burst read of 3 32-bit words) and Byte Count (a single 32-bit read).

2.8.3 Monitoring (Sniffer) Mode

The CPU can program the GT-48002 to work in Monitoring mode in one of its ports. This is done by setting bit 2 in the Port Control register. In this mode, the GT-48002 sends all receive (including local traffic) and transmit packets to the CPU or to a port in one of the GT-48001/2 devices which was assigned to be the target Sniffer. To assign a port to be the target Sniffer, you must:

1. Write the target Sniffer Device Number and Port Number in all GT-48001/2 devices' CPU and Sniffer Numbers Register (Offset: 0x140030).

2. Set bit 9 of all GT-48001/2 devices' Global Control Register to 1 to indicate a GT-48001/2 port will be used as the Sniffer.

3. Enable Sniffer mode in the target Sniffer device(s) ONLY by setting bit 2 in the Port Control Register.

The packets that are forwarded to the Sniffer are not in a linear order.

2.8.4 Spanning Tree Support

The GT-48002 provides the required hardware assistance for Spanning Tree algorithm implementation. The Spanning Tree algorithm is performed by the CPU.

The GT-48002 holds a SpanEn bit in the Global Control register and additional SpanEn bits in each of the Port Control registers. The following table summarizes the hardware assistance for the Spanning Tree algorithm and the treatment of Bridge Protocol Data Units (BPDU).

SpanEn (Global)	SpanEn (Port)	Logic State	Remarks
0	x	Port Enable	No Spanning Tree. Treat BPDU as regular Multicast
1	1	Blocking, Listening, Learning	Transfer BPDUs to the CPU. All receive/transmit packets should be rejected. Accept BPDUs from the CPU. No address learning
1	0	Forward	Transfer BPDUs to the CPU. Accept all packets. Address learning

The GT-48002 does not learn during the traditional 'Learning' stage. It only starts learning during the Forward stage.

2.9 Serial Interfaces

The GT-48002 interfaces directly to two MII (Media Independent Interface) ports which are compliant with the IEEE 802.3u Fast Ethernet standard. Each MII port has the following characteristics:

- Capable of supporting both 10 Mb/s and 100 Mb/s data rates in half or full duplex modes
- Data and delimiters are synchronous to clock references
- Provides independent 4-bit wide transmit and receive paths
- Uses TTL signal levels
- Provides a simple management interface (common to all ports)
- Capable of driving a limited length of shielded

cable

2.9.1 10/100 Mbps MII Compatible Interface

The GT-48002 MAC allows it to be connected to a 10 or 100Mbps network. The GT-48002 interfaces to an IEEE 802.3 10/100 Mbps MII compatible PHY device. The data path consists of a separate nibble-wide stream for both transmit and receive activities. The GT-48002 can switch automatically between 10 or 100 Mbps operation depending on the speed of the network. Data transfers are clocked by the 25 MHz transmit and receive clocks in 100 Mbps operation, or by 2.5 MHz transmit and receive clocks in 10 Mbps operation. The clock inputs are driven by the PHY, which controls the clock rate based on auto-negotiation.

The GT-48002 MAC performs all of the functions of the 802.3 protocol such as frame formatting, frame stripping, collision handling, deferral to link traffic, etc. The GT-48002 ensures the any outgoing packet complies with the 802.3 specification in terms of preamble structure. The GT-48002 transmits 56 preamble bits before Start of Frame Delimiter (SFD). The MAC can also be placed in a Full-duplex mode which allows for simultaneous transmission and reception of frames.

2.9.2 10/100 Mbps MII Transmission

When the GT-48002 has a frame ready for transmission, it samples the link activity. If the CrS signal is inactive (no activity on the link), and the Inter Frame Gap (IFG) counter has expired, frame transmission begins. The IFG default value is 96 bit times, and can be programmed by the CPU. The data is transmitted via pins TxD[3:0] of the transmitting port, clocked on the rising edge of TxClk. The signal TxEn is asserted at this same time. In the case of collision, the PHY asserts the CoL signal on the GT-48002 which will then stop transmitting the frame and append a jam sequence onto the link. After the end of a collided transmission, the GT-48002 will back off and attempt to retransmit once the backoff counter expires.

The GT-48002 implements the truncated exponential backoff algorithm defined by the 802.3 standard. After 16 consecutive retransmit trials, the GT-48002 resets its collision counter and restarts the backoff algorithm, and continues to try and retransmit the frame. The retransmission is done from the data already stored in the DRAM. In the case of a successful transmission, the GT-48002 is ready to transmit any other frames queued in its transmit FIFO within the minimum IFG of the link.

The GT-48002 implements a programmable data blinder with default value of 32 bits before the end of the IFG. The IFG counter starts to time the interframe gap as soon as transmitting and carrier sense are both false. When timing the interframe gap, the IFG counter is reset if carrier is detected during the first part of the interframe gap before entering the data blinder zone. During the data blinder zone, the IFG counter will not be reset to ensure fair access to the medium.

Any one of the GT-48002 MII ports can be automatically partitioned under excessive duration or frequency of collision conditions, if the partition function is enabled in the GT-48002 control register (the default is partition disabled). The GT-48002 will continue to transmit data packets to a partitioned port, but will not respond to activity on the partitioned port's receive lines. The GT-48002 will continue to monitor the port and reconnect it once certain criteria indicating port 'wellness' are met. The criteria for partition/reconnection are specified by the 802.3 standard. Each GT-48002 port, is partitioned and/or reconnected separately and independently of other network ports.

2.9.3 10/100 Mbps MII Reception

Frame reception starts with the assertion of CrS (while the GT-48002 is not transmitting) by the PHY. Once RxDV is asserted, the GT-48002 will begin sampling incoming data on pins RxDV[3:0] on the rising edge of RxClk. Reception ends when the RxDV is deasserted by the PHY. The last nibble sampled by the GT-48002 is the nibble present on RxD[3:0] on the last RxClk rising edge in which RxDV is still asserted. During reception, the RxDV is asserted. If, while RxDV is asserted, the GT-48002 detects the assertion of RxEr, it will designate this frame as a corrupted frame. While no reception is taking place, RxDV should remain deasserted.

2.9.4 10/100 Mbps Full-Duplex Operation

When operating in Full-duplex mode the GT-48002 can transmit and receive frames simultaneously. In full-duplex mode, the CrS signal is associated with received frames only and has no effect on transmitted frames. The Col signal is ignored by the GT-48002 while in Full-duplex mode. Transmission starts when TxEn goes active. Transmission starts regardless of the state of CrS. Reception starts when the CrS signal is asserted indicating traffic on the receive port of the PHY.

2.9.5 MII Management Interface (SMI)

The GT-48002 MAC contains an MII Management Interface (SMI) to an MII compliant PHY. This allows control and status parameters to be passed between the GT-48002 and the PHY (parameters specified by the CPU) by one serial pin (MDIO) and a clocking pin (MDC), reducing the number of control pins required for PHY mode control. Typically, the GT-48002 will continuously query the PHY devices for their link status, without CPU intervention. The predefined PHY addresses for the link query are 1 and 2 (out of possible 32 addresses). This protocol conforms with the National DP83840 PHY device as well as other available PHYs.

A CPU connected to the GT-48002 can Write/Read to/from all PHY addresses/registers, by writing and reading to/from a dedicated set of GT-48002 SMI control registers. The SMI allows the CPU to have direct control over an MII-compatible PHY device via the GT-48002 SMI control register. This allows the driver software to place the PHY in specific modes such as Full Duplex, Loopback, Power Down, 10/100 speed selection as well as control of the PHY device's Auto Negotiation function if it exists. The CPU writes commands to the GT-48002 SMI register and the GT-48002 reads or writes control/status parameters to the PHY device via a serial, bi-directional data pin called MDIO. These serial data transfers are clocked by the GT-48002 MDC clock output.

2.9.6 SMI Cycles

The SMI protocol consists of a bit stream that is driven or sampled by the GT-48002 on each rising edge of the MDC clock. The bit stream format of the SMI frame is described in the table below:

	PRE	ST	OP	PhyAd	RegAd	TA	Data	IDLE
RD	*1...1*	*01*	*10*	*AAAAA*	*RRRRR*	*Z0*	*D..D(32)*	*Z*
WR	*1...1*	*01*	*01*	*AAAAA*	*RRRRR*	*10*	*D..D(32)*	*Z*

PRE (Preamble) - At the beginning of each transaction, the GT-48002 sends a sequence of 32 contiguous logic one bits on MDIO with 32 corresponding cycles on MDC to provide the PHY with a pattern that it can use to establish synchronization.

ST (Start of Frame) - A Start of Frame pattern of 01

OP (Operation Code) - 10 - READ; 01 - WRITE

PhyAd (PHY Address) - A 5 bit address of the PHY device, allowing 32 unique PHY addresses. The first PHY address bit transmitted by the GT-48002 is the MSB of the address.

RegAd (Register Address) - A 5 bit address of the PHY register, allowing 32 individual registers to be addressed withing each PHY. The first register address bit transmitted by the GT-48002 is the MSB of the address. The GT-48002 always queries the PHY device for status of the link by reading register #00001 bit #2.

TA (TurnAround) - The turnaround time is a 2 bit time spacing between the Register Address field and the Data field of the SMI frame to avoid contention during a READ transaction. During a READ transaction the PHY should not drive MDIO in the first bit time and drive '0' in the second bit time. During a WRITE transaction, the GT-48002 drives a '10' pattern to fill the TA time.

Data (Data) - The data field is 16 bits long. The PHY drives the data field during READ transactions. The GT-48002 drives the data field during WRITE transactions. The first data bit transmitted and received shall be but 15 of the PHY register being addressed.

IDLE (Idle) - The IDLE condition on MDIO is a high impedance state. The MDIO driver is disabled and the PHY should pull-up the MDIO line to a logic one.

2.9.7 Link Detection and Link Detection Bypass (ForceLinkPass*)

Typically, the GT-48002 will continuously query the PHY devices for their link status, without CPU intervention. The predefined PHY addresses for the link query are 1 and 2 (out of possible 32 addresses). The GT-48002 will alternately read register#1 from PHY#1 and PHY#2 and update the internal link bits according to the value of bit#2 of register#1. In the case of "link is down" (i.e. bit#2 is '0'), that port will enter link test fail state. In this state, all of the port's logic is reset. The port will exit from link test fail state only when the "link is up" i.e. bit#2 of register#1 is read from the port's PHY as '1'.

The GT-48002 offers the option to disable this link detection mechanism by forcing the link state of both ports to the link test pass state. This is done by connecting the ForceLinkPass* input to a pull-down resistor. This pin is sampled at reset. When connected to pullup, The link status of the ports is read through the SMI from the PHY devices (register#1, bit#2). When connected to a pull-down, the link status of all ports remains in the "link is up" state regardless of the PHY's link bit value.

2.9.8 LED Support

2.9.8.1 LED Interface General Description

The GT-48002 supports two types of LED interfaces: a serial interface and a parallel interface. The serial LED interface is similar to the 3-pin LED interface of the GT-48001 device. The parallel interface offers the LED information directly on the device outputs. An external driver is required to drive the LEDs. The GT-48002 supports two modes of port status information display selectable via the LEDMode input.

The basic accessable data via the LED interface is:

- Port Status (two modes of operation)
- Transmit data in progress (TxEn)
- Receive data in progress (RxDV)
- Collision active (Col)
- Forwarding of Unknown packets enabled
- Port configured as Sniffer
- Full/Half Duplex
- Receive Buffer Full

2.9.8.2 Detailed LED Signal Description

1. Primary Port Status LED: Port Status LED indicates the port status in two operation modes selectable via the LEDMode input:

1.1. Primary Port Status LED in Mode 0: (LEDMode input is LOW)

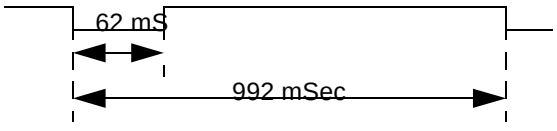
At this mode, the Port Status LED provides the following information:

- if Port is disabled - Port Status LED is OFF
- else if Link Integrity test failed - Port Status LED blinks once
- else if Partition State detected - Port Status LED blinks twice
- else Everything is OK - Port Status LED is ON

Status LED blink timing (MODE 0)

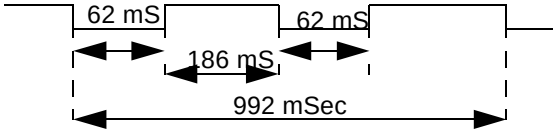
Link Integrity test failed - Status LED blinks once

* status bit is active for 62 mS every 992 mS



Partition - Status LED blinks twice

* status bit is activated twice every 992 mS for 62 mS each time, with a period of 186 mS between two consequtive activations.



1.2. Primary Port Status LED in Mode 1: (LEDMode input is HIGH)

At this mode, the Port Status LED initially displays the port link status information (indication type A) and then switches to reflect the port trafic (Transmit or Receive) activity (indication type B). The switching between the two types of indications is as follows:

Indication type A: The Port Status LED indicates the port link status for a period of about 3 to 3.5 seconds (active - link is up, inactive - link is down) following any of the events below:

- RESET deassertion
- transition from link down to link up

Following this time period, the Port Status LED will switch to indication type B. In the case that the link is down, the Port Status LED will remain inactive and will not switch to indication type b.

Indication type b: The Port Status LED indicates the port trafic (Transmit or Receive) activity which is a logical OR of the TxEn active and RxDV active dynamic signals. The "monostable" function is applied to this indication type so the LED can be viewed for a period of about 62 mS for LedMode=0 or for about 7.5 mS for LedMode=1, per each trafic activity. The Port Status LED will switch to indication type a on the following cases:

- RESET assertion
- transition from link up to link down

3. Transmit data in progress (TxEn): This signal indicates the port transmit activity - (TxEn output is active)

4. Receive data in progress (RxDV): This signal indicates the port receive activity - (RxDV input is active)

5. Collision active (Col): This signal indicates the port collision event - (Col input is active)

6. Full/Half# duplex: This signal indicates the port duplex: active - full duplex, inactive - half duplex.

7. Receive Buffer Full: This signal indicates the port receive buffer status: active - the buffer exceeds it's pro-

grammed threshold, inactive otherwise.

8. orwarding of unknown packets enabled: This signal indicates the port mode of forwarding unknown packets: active - forwarding unknown packets enabled, inactive otherwise

9. The port is configured as Sniffer: This signal indicates if the port mode is configured as a sniffer target port: active - port is a target sniffer, inactive otherwise

10. Link Fail State: This signal indicates the port link status: active - link is down, inactive - link is up.

11. Partition State: This signal indicates the port partition status: active - port entered partition state, inactive otherwise.

12. Secondary Port Status LED: This signal indicated the secondary status mode as per the inverted value of LEDMode input.

13. Pure Port Status LED: This signal will be inactive for any of the following events: Port is disabled -or- Link Integrity test failed -or- Partition State detected; active otherwise.

2.9.8.3 LED Signals Timing Type

1. **Static LED Signals**: These signals are stable for relatively long time periods. The LED indication directly reflects their current value. The static signals are:

- Port Status (mode 0)
- Forwarding of Unknown packets enabled
- Port configured as Sniffer
- Full/Half Duplex

2. **Dynamic Internal Signals**: These signals are typically active for short time periods. In order to be visible through the LED Indication Interfaces, the GT-48002 includes a "monostable" function per each of these dynamic signals so they can be viewed on the LED indication output for an extended period of time selectable via the led_mode input: led mode 0: about 62 mS; led mode 1: about 7.5 mS. The dynamic signals are:

- Port Status (mode 1)
- Transmit data in progress (TxEn)
- Receive data in progress (RxDV)
- Collision active (Col)
- Receive Buffer Full

2.9.8.4 LED Indications Serial Interface Description

The LED Serial Indications Interface consists of three outputs:

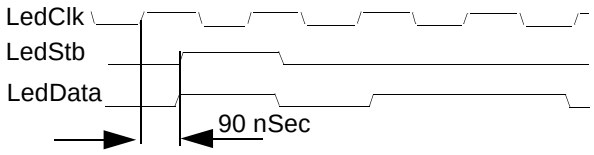
- LEDClk : primary clock
- LEDStb : start of data frame indication (active HIGH)
- LEDData : multiplexed data output (active LOW)

LEDClk: LEDClk is the primary timebase of the LED Indications Interface. It is a 50% duty cycle free running clock at a fixed frequency selectable via the led_mode input: led mode= 0: 1MHz; led mode= 1: 202 KHZ. LED_CLK is active when Rst* is asserted. LEDClk frequency during Rst* is 33 MHz.

LEDStb: LEDStb (active HIGH) indicates the begining of the data frame. LEDStb is activated for a duration of one LEDClk cycle once every 128 LEDClk cycles, starting from Rst* deactivation. This signal marks the beginning of the 128 bit long LED data frame. LEDStb transitions occur 90 nS after LEDClk rising edge.

LEDData: The internal signals are multiplexed on the LEDData output for every data frame. LEDStb activation signals the presence of data bit #1 on the LEDData output. LEDData transitions occur 90 nS after LEDClk rising edge. All internal signals are active HIGH. Note that LEDData is active LOW. This means that when the internal indication listed in table 3.4 below, is active, LEDData is LOW for the corresponding bit. For example: is bit #9 (txen_active[0]) in the LEDData serial stream is LOW, this indicates that port #0 is transmitting data.

2.9.8.5 LED Indications Serial Interface Signals - Timing Diagram



2.9.8.6 Table of Internal Activities/Status Driven via the LED Serial Indications Interface

The following table contains a bit by bit description of the internal signals driven through the LED Indications Interface. The bit number (bit#) refers to the activation of LedStb. LedStb is active for bit# 1. PRIVATE bits contents are not defined (i.e. can be either HIGH or LOW).

bit#	Symbolic Signal Name [port number]
1	primary_port_status[0]
2	primary_port_status[1]

bit#	Symbolic Signal Name [port number]
3-8	PRIVATE
9	txen_active[0]
10	rxdv_active[0]
11	collision[0]
12	rx_buffer_full[0]
13	unknown_enable[0]
14	port_sniffer[0]
15	full_duplex[0]
16	PRIVATE
17	txen_active[1]
18	rxdv_active[1]
19	collision[1]
20	rx_buffer_full[1]
21	unknown_enable[1]
22	port_sniffer[1]
23	full_duplex[1]
24-72	PRIVATE
73	link_test_fail[0]
74	link_test_fail[1]
75-80	PRIVATE
81	partition[0]
82	partition[1]
83-88	PRIVATE
89	secondary_port_status[0]
90	secondary_port_status[1]
91-96	PRIVATE
97	pure_port_status[0]
98	pure_port_status[1]
99-128	PRIVATE

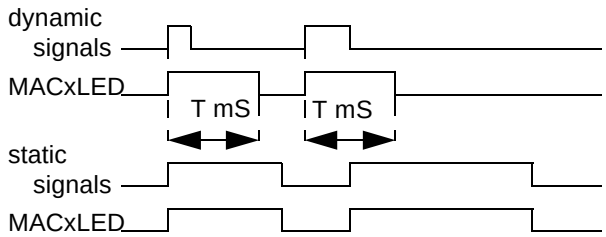
2.9.8.7 LED Indications Parallel Interface Description

The LED Parallel Indications Interface consists of the MACLED output pins per port (port 0 - MAC0LED; port 1 - MAC1LED):

- MAC0LED[0],MAC1LED[0] : Port Status LED
- MAC0LED[1],MAC1LED[1] : Transmit data in progress (TxEn)
- MAC0LED[2],MAC1LED[2] : Receive data in progress (RxDV)
- MAC0LED[3],MAC1LED[3] : Collision active (Col)
- MAC0LED[4],MAC1LED[4] : Full/Half# duplex
- MAC0LED[5],MAC1LED[5] : Receive Buffer Full

The MAC0LED and MAC1LED outputs are active LOW. An external buffer should be added per each driven LED. The MAC0LED and MAC1LED outputs reflect the same data (value and timing) which is accessable via the Serial Interface. The Parallel Interface includes the same "monostable" function on the Dynamic Internal Signals as the Serial Interface. The monostable" function timing is selected via the LEDMode input.

2.9.8.8 LED Indications Parallel Interface Signals - Timing Diagram



where: T = ~62 mS for LEDMode input = 0

T = ~7.5 mS for LEDMode input = 1

2.10 PCI Interface

The GT-48002 interfaces directly with the PCI bus. It can be either a master initiating a PCI bus operation or a target responding to a PCI bus operation. All GT-48002 internal registers and addresses are memory mapped. Bit 21 in the address selects between the DRAM area and the internal registers. The GT-48002's DRAM is accessible by the CPU.

All PCI transactions are served in round-robin scheme. These transactions constitute a key element of the GalNet architecture, and consist of 5 basic messages:

1. **'new_address'**: A message between GT-48002 devices or between the CPU and a GT-48002 device that contains information about a new MAC Address. This message is used also by the CPU to update the Address

Table. The message format is as follows:

PCI Bits	Description
Address	
[31:27]	'00001'
[26:22]	Target Device Number
[21]	'0' (Internal registers)
[20:18]	'100' ('new_address' message)
[17:0]	'0'
Data 0	
[31:3]	MAC address [19:47]
[2]	Aging
[1]	Skip
[0]	Valid
Data 1	
[31]	Always '1'
[30]	Static
[29]	AddressUnknown/New address
[28]	Multiple
[27]	Always '0'
[26:24]	Port#
[23:19]	Device#
[18:0]	MAC address [0:18]
Data 2	
[24]	Intervention mode for Destination Address
[25]	Intervention mode for Source Address

2. **'buffer_request'**: A message from the source device to the target device to request a buffer. The format is as follows:

PCI Bits	Description
Address	
[31:27]	'00001'
[26:22]	Target Device Number
[21]	'0' (Internal registers)
[20:18]	'101' ('buffer_request' message)
[17:13]	Source Device Number
[12:0]	'0'
Data 0	
[31]	Sniffer (0-Sniffer type message)
[30]	Unknown (0-Unknown message)
[29:28]	'0'
[27:25]	Source Port#
[24:22]	Target Port#
[21]	Multicast/Unicast (0-Unicast)
[20:10]	Byte Count
[9:0]	Source Buffer Address (divided by 0x600)

3. **'start_of_packet'**: A message from a target device to the source device which contains the Empty Buffer address. The format is as follows:

PCI Line	Description
Address	
[31:27]	'00001'
[26:22]	Source Device Number
[21]	'0' (Internal registers)
[20:18]	'110' ('start_of_packet' message)
[17:0]	'0'
Data 0	
[31]	Sniffer (0 - Sniffer type message)
[30]	'0'
[29:22]	Target Port# (1bit per each port; bit 22 - Port 0, bit 23 - Port 1 etc.)
[21]	Multicast/Unicast
[20:10]	Byte Count
[9:0]	Target Buffer Address (divided by 0x600)
Data 1	
[31:18]	'0'
[17:15]	Source Port#
[14:5]	Source Buffer Address (divided by 0x600)
[4:0]	Target Device Number

4. **'packet_transfer'**: A burst of 8 32-bit words from the source device to the target device which contains the packet. The format is as follows:

PCI Line	Description
Address	
[31:27]	'00001'
[26:22]	Target Device Number
[21]	'1' (DRAM)
[20:0]	DRAM location
Data 0	
[31:0]	Data 0
----	----
----	----
Data 7	
[31:0]	Data 7

5. **'end_of_packet'**: A message from the source device to the target device which indicates the end of the packet. The format is as follows:

PCI Line	Description
Address	
[31:27]	'00001'
[26:22]	Target Device Number
[21]	'0' (Internal registers)
[20:18]	'111' ('end_of_packet' message)
[17:0]	'0'
Data 0	
[31:30]	'0'
[30]	Unknown
[29:22]	Target Port# (1bit per each port; bit 22 - Port 0, bit 23 - Port 1 etc.)
[21]	Multicast/Unicast
[20:10]	Byte Count
[9:0]	Target Buffer Address (divided by 0x600)

2.11 DRAM Interface

The GT-48002 supports EDO DRAMs. The usable bandwidth in EDO mode satisfies the required bandwidth for data transfer, address recognition and Tx descriptor fetch/update.

2.12 Reset Configuration

The GT-48002 must acquire some knowledge about the system during initialization. Certain pins must be pulled up or down externally to accomplish this. The recommended value of the resistors is 4.7K ohms. The GT-48002 samples these input signals during reset.

Pin		Configuration Function
DAddr[4:0]		Device Number (bit 4 is MSB) **Note**: For unmanaged operation (CPU is disabled) the Device Number should be different from zero.
DAddr[5]		DRAM Size
	0-	2Mbyte
	1-	1Mbyte
DAddr[6]		Half/Full Duplex Mode for Port 0
	1-	Full duplex
	0-	Half duplex
DAddr[7]		Half/Full Duplex Mode for Port 1
	1-	Full duplex
	0-	Half duplex
DAddr[8]		DRAM Type
	1-	EDO
ForceLinkPass*		Force Link Pass
	1-	Read link status from PHY via SMI
	0-	Force link status to "link is up" for all ports

3 REGISTER TABLES

The GT-48002 incorporates the required PCI Configuration registers, Command registers and various counters for management purposes. The GT-48002 can work in stand-alone mode, in which there is no requirement for CPU intervention (a system with no CPU) and the default values of the control registers are used.

3.1 Register Address Calculation

The actual address of an internal register is the sum of the GT-48002 base address and the particular register's Offset. The CPU defines the base address by writing to the Internal Registers Base Address register in the PCI Configuration area.

REGISTER ADDRESS [31:27] = 00001

REGISTER ADDRESS [26:22] = Device Number (bit 26 is MSB)

REGISTER ADDRESS [21:0] = Register Offset

3.2 Register Map

Description	Register Offset
Internal Registers	
Global Control	0x140028
Port Control 0	0x040200
Port Control 1	0x040204
Status	0x14002c
CPU and Sniffer Numbers	0x140030
Interrupt Cause	0x044
Interrupt Mask	0x048
Serial Parameters	0x040220
Rx Buffers Threshold	0x040224
CPU Buffer Base Address	0x140034
CPU Start Of Packet Base Address	0x140038
CPU New Address Base Address	0x14003c
CPU Intervention Base Address	0x140048
Device Table	0x40
Timeout Counter	0x04c
Port 0 Counter Block	0x040000 - 0x040058
Port 1 Counter Block	0x040080 - 0x0400c8
PCI Global Counters	0x140040- 0x140044
SMI Register	0x14004c
PCI Configuration	
Device and Vendor ID	0x000
Status and Command	0x004

Class Code and Revision ID	0x008
Header Type, Latency Timer, Cache Line	0x00c
DRAM Base Address	0x010
Internal Registers Base Address	0x014
Interrupt Pin and Line	0x03c

3.3 Internal Registers

Global Control, Offset: 0x140028

Bits	Field Name	Function	Initial Value
0	DisLearnPro	Disable Learning Process. 0 - Learning process is enabled 1 - Learning process is disabled. The GT-48002 will not learn any new addresses from the wire	0x0
1	RMONEn	RMON Enable. 0 - RMON disabled 1 - The GT-48002 enters the RMON mode (Station-to-Station connectivity matrix) and asserts the ChipSel* pin when it reads the packet's Byte Count and Source and Destination Addresses from the DRAM	0x0
2	DevTabMod	Device Table Mode. 0 - The GT-48002 updates the appropriate bits in the table upon master abort 1 - The CPU updates the Device Table	0x0
3	-	Reserved.	0x1
4	DRAMArbPri	DRAM Arbiter Priority. This bit indicates the DRAM arbitration scheme of the four GT-48002 internal units as follows: 0 - 1) Frame Control unit, 2) Switching Core unit, 3) PCI and InterPCI Control units in round robin scheme. 1- 1) Frame Control unit, 2) Switching Core unit, 3) InterPCI Control unit, 4) PCI Control unit	0x0
5	DescArbPri	Descriptor Control Arbiter Priority. This bit indicates the descriptor control arbitration scheme as follows: 0 - Round robin between the PCI side and the 2 Ethernet ports. The 2 ports have equal priority. 1- PCI has higher priority than the 2 ports. The priority is: PCI, Port 0; PCI, Port 1; PCI, Port 0; PCI, Port 1...	0x1
6	ForwUnk	Forward Unknown. It defines whether the GT-48002 will forward Unknown packets to the CPU or not. 0 - Do not forward 1 - Forward	0x0

Bits	Field Name	Function	Initial Value
7	ForwNewAdd	Forward New Address. It defines whether the GT-48002 will forward new address messages to the CPU or not. 0 - Do not forward 1 - Forward	0x0
8	RecEn	Recovery Enable. It defines whether the recovery process is enabled or not. 0 - Disabled 1 - Enabled	0x1
9	SnifTyp	Sniffer Type. This bit indicates the Sniffer type. The Sniffer can be a CPU or a dedicated port in one of the GT-48002 devices which was assigned to be the Sniffer. 0 - CPU type 1 - GT-48002 type	0x1
10	CPUEn	CPU Enable. This bit indicates that there is a CPU in the system. 0 - CPU does not exist 1 - CPU exists	0x0
11	RMONToPCI	RMON to PCI Enable. Meaningful only when RMONEn bit is set. 0 - The GT-48002 reads the DA/SA of the packet that is forwarded only to the local ports. 1 - The GT-48002 reads the DA/SA of the packet that is forwarded to the local ports and the PCI.	0x0
19:12	-	Reserved.	0x0
20	BufThrEn	Buffer Threshold Enable. 0 - There is no limitation on the buffers' allocation. 1 - The buffers allocated to the ports and the PCI are limited to the number which is written in the Rx Buffers Threshold register.	0x1
21	-	Reserved.	0x0
22	ForwMulti	Forward Multicast. 0 - The GT-48002 forwards Multicast packets to all the ports. 1 - Multicast packets forwarded only to the CPU.	0x0
23	ParEn	Partition Enable. When more than 64 collisions occur while transmitting, the GT-48002 enters the Partition mode. It waits for the first good packet from the wire, and then exits Partition mode. At Partition mode, the port continues transmision, but ignores incoming packets. 0 - Disable Partition mode 1- Enable Partition mode	0x0

Bits	Field Name	Function	Initial Value
24	SpanEn	Spanning Tree Enable. 0 - The BPDU (Bridge Protocol Data Unit) packets are treated as Multicast packets, and therefore are forwarded to all ports. 1 - The GT-48002 forwards BPDU packets only to the CPU.	0x0
25	-	Reserved.	0x0
26	CountMode	MIB Counters mode 0 - Counter mode 0 1 - Counter mode 1	0x0
27	EnDevSw	Enable Device Software Control 0- No effect (Enable Device controlled by EnDev* input) 1- Same as deasserting EnDev* input. The ports and the PCI are disabled.	0x0
28	RstQueueSw	Reset Transmit Queues Software Control 0- No effect (Reset Transmit Queues controled by RstQueue* input) 1- Same as asserting RstQueue* input. All internal transmit and receive queues are cleared. All GT-48002 state machines are moved to their initial state.	0x0

Port Control (2 Registers), Offset: 0x040200 - 0x040204

Bits	Field Name	Function	Initial Value
0	PortEn	Port Enable. 0 - Port is disabled 1 - Port is enabled	0x1
1	FullDx	Half/Full Duplex. 0 - Port works in half-duplex mode 1 - Port works in full-duplex mode **Note**: this bit may be changed only when PortEn bit is set to 0 (Port is disabled).	DAddr[6] for port0 or DAddr[7] for port1 (value sampled at reset)
2	MonMode	Monitoring Mode. 0 - Port works in normal mode 1 - Port works in monitoring mode; all Rx and Tx packets are sent to the Sniffer Target	0x0
5:3	Reserved	Reserved.	0x1
6	FilBroad	Filter Broadcast. 0 - Broadcast packets are forwarded to all ports. 1 - The GT-48002 discards Broadcast packets.	0x0
7	ForwUnk	Forward Unknown. 0 - Unknown packets are forwarded. 1 - The GT-48002 does not forward Unknown packets to this port.	0x0

Bits	Field Name	Function	Initial Value
8	SpanEn	Spanning Tree Enable. Meaningful only when SpanEn bit in the Global Control register is set. 0 - All packets are accepted. 1 - The GT-48002 discards all incoming/outgoing packets except for BPDU packets.	0x0
9	LongEn	Long Packets Enable. 0 - All packets with max FrameSize=1518 bytes are accepted. 1 - All packets with max FrameSize=1522 bytes are accepted (may be used for VLAN tagging).	0x0

Status, Offset: 0x14002c (Read-only register)

Bits	Field Name	Function	Initial Value
4:0	DevNum	Device Number. Indicates the GT-48002 number chosen by the designer.	DAddr[4:0] sampled at reset bit 4 is MSB
5	DRAMSize	DRAM Size. Indicates the DRAM size. 0- 2Mbyte 1- 1Mbyte	DAddr[5] sampled at reset
6	Port0Par	Port0 Partition. This bit indicates the port 0 Partition status. 0 - No Partition (Normal mode) 1 - Partition	0x0
7	Port0LTF	Port0 Link Test Fail. This bit indicates the port 0 Link Test status. 0 - Link Test Pass (link is up) 1 - Link Test Fail (link is down)	0x0
8	Port1Par	Port1 Partition. This bit indicates the port 1 Partition status. 0 - No Partition (Normal mode) 1 - Partition	0x0
9	Port1LTF	Port1 Link Test Fail. This bit indicates the port 1 Link Test status. 0 - Link Test Pass (link is up) 1 - Link Test Fail (link is down)	0x0

CPU and Sniffer Numbers, Offset: 0x140030

Bits	Field Name	Function	Initial Value
4:0	SnifDevNum	Sniffer Device Number. These bits specify the Sniffer Device Number (Target) chosen by the designer. **Note**: The same value should be programmed to all GT-48002/GT-48001 devices sharing the same PCI bus.	0x0 (bit 4 is MSB)
7:5	SnifPortNum	Sniffer Port Number. These bits specify the Sniffer Port Number (Target) chosen by the designer. **Note**: The same value should be programmed to all GT-48002/GT-48001 devices sharing the same PCI bus.	0x0 (bit 7 is MSB)
12:8	MgmtDevNum	Management Device Number. These bits specify the Management Device Number chosen by the designer. **Note1**: The same value should be programmed to all GT-48002/GT-48001 devices sharing the same PCI bus. **Note2**: The corresponding bit for the MgmtDevNum value should be set in the Device table (offset 0x40) for all GT-48002/GT-48001 devices sharing the same PCI bus.	0x0 (bit 12 is MSB)

Interrupt Cause, Offset: 0x044

Bits	Field Name	Function	Initial Value
0	IntSumm	Interrupt Summary. Logical ‘OR’ of all the interrupt bits.	0x0
1	NewAdd	New Address. This bit is set by the GT-48002 when a new address is received.	0x0
2	TxEnd	Tx End of Packet. This bit is set by the GT-48002 upon transferring a ‘end_of_packet’ message to the CPU main memory	0x0
3	RxStart	Rx Start of Packet. This bit is set by the GT-48002 upon sending a ‘start_of_packet’ message to the CPU	0x0
4	AddrRecF	Address Recognition Failed. This bit is set by the GT-48002 when the address recognition cycle fails (due to a large number of MAC addresses).	0x0
5	FlushTxQ	Flush Tx Queue. This bit is set by the GT-48002 when one of the Tx queues is flushed due to the Watchdog Timer.	0x0
6	MastRdPar	Master Read Parity. This bit is set by the GT-48002 upon master read parity error on the PCI.	0x0

Bits	Field Name	Function	Initial Value
7	MastWrPar	Master Write Parity. This bit is set by the GT-48002 upon master write error on the PCI.	0x0
8	AddPar	Address Parity. This bit is set by the GT-48002 upon address parity error on the PCI.	0x0
9	MastAbort	Master Abort. the This bit is set by the GT-48002 upon master abort on PCI.	0x0
10	TarAbort	Target Abort. This bit is set by the GT-48002 upon target abort on the PCI.	0x0
11	LTF	Link Test Fail. This bit is set by the GT-48002 upon Link Test Fail in one of the ports.	0x0
12	Part	Partition. This bit is set by the GT-48002 upon entering Partition state in one of the ports.	0x0
13	BufWrap	Buffer Wrap-Around. This bit is set by the GT-48002 upon transferring 16 packets to the CPU main memory.	0x0
14	Interv	Intervention. This bit is set by the GT-48002 upon transferring a ‘buffer_request’ message in Intervention mode.	0x0
15	IntervWrap	Intervention Wrap-Around. This bit is set by the GT-48002 upon transferring 256 ‘buffer_request’ messages in Intervention mode.	0x0

Interrupt Mask, Offset: 0x048

Bits	Field Name	Function	Initial Value
15:0	MaskBits	Mask to the CPU interrupt line for the appropriate bits in the Interrupt Cause register.	0x0

Serial Parameters, Offset: 0x040220

Bits	Field Name	Function	Initial Value
11:0	-	Reserved.	0x813
16:12	IFGData	Inter-Frame Gap (IFG): The IFG varies between 12 bit-times (0x3) and 124 bit-times (0x1F). The step is 40ns (4 bit-times). The default value is 18 hex, or 0.96us. Value should be written in hexadecimal format. **Note**: these bits may be changed only when PortEn bits are set to 0 in all Port Control Registers (Port is disabled).	0x18 = 24 decimal

Bits	Field Name	Function	Initial Value
21:17	DataBlinder	Data Blinder: The number of nibbles from the beginning of the IFG, in which the GT-48002 will restart the IFG counter when detecting a carrier activity. Following this value, the GT-48002 will enter the Data Blinder zone and will not reset the IFG counter to ensure fair access to the medium. Value should be written in hexadecimal format. The default is 10 hex (64 bit times - 2/3 of the default IFG). The step is 40ns (4 bit-times). Valid range is 3 to 1F hex nibbles. **Note**: these bits may be changed only when PortEn bits are set to 0 in all Port Control Registers (Port is disabled).	0x10 = 16 decimal
25:22	-	Reserved.	0xb

Rx Buffers Threshold, Offset: 0x040224

Bits	Field Name	Function	Initial Value
3:0	TxWatTim	Tx Watchdog Timer. For 100 Mbps operation, the default value of the timer is 63msec and the range is between 10.5mSec and 168msec, in 10.5 mSec steps. For 10 Mbps operation, the default value of the timer is 630msec and the range is between 105mSec to 1680msec, in 105 mSec steps. Valid values: 1 to F hex. Value should be written in hexadecimal format.	0x6 (63 mS @ 100Mbps; 630 mS @ 10 Mbps)
8:4	-	Reserved.	0x04
12:9	RxBufThr	Receive Buffer Threshold. These bits determine the threshold of the receive buffers. Meaningful only when the BufThrEn bit in the Global Control register is set. The Receive Buffer Threshold equals to: (RxBufThr x 20 + 20), the step is 20, and the value varies between 20 (0x0) and 320 (0xf). The default is 80 (0x3) for 1MB DRAM and 160 (0x7) for 2MB DRAM. Value should be written in hexadecimal format.	0x3 (80) @ 1M DRAM 0x7 (160) @ 2M DRAM
15:13	-	Reserved.	0x0
18:16	PCIBufThr	PCI Buffer Threshold. These bits determine the threshold of the PCI receive buffers. The PCI Buffer Threshold equals to: (PCIBufThr x 50 +100). step is 50, and the value varies between100 (0x0) and 320 (0xf). The default is 150 (0x1) for 1MB DRAM and 200 (0x2) for 2MB DRAM. Value should be written in hexadecimal format.	0x1 (150) @ 1M DRAM 0x2 (200) @ 2M DRAM

CPU Buffer Base Address, Offset: 0x140034 (a second write updates the Shadow register)

Bits	Field Name	Function	Initial Value
31:15	BaseAdd	Contains a pointer to the CPU buffer area	0x0
14:0	-	Reserved.	-

CPU Start of Packet Base Address, Offset: 0x140038

Bits	Field Name	Function	Initial Value
31:8	StBaseAdd	Contains a pointer to the CPU 'start_of_packet' area. The area includes 32 entries (1 32-bit word each) for the GT-48002's 'start_of_packet' messages.	-
7:0	-	Reserved.	-

CPU New Address Base Address, Offset: 0x14003c

Bits	Field Name	Function	Initial Value
31:8	NABaseAdd	Contains a pointer to the CPU 'new_address' area. The area includes 32 entries (2 32-bit words each) for the GT-48002's 'new_address' messages.	0x0
7:0	-	Reserved.	-

CPU Intervention Base Address, Offset: 0x140048 (a second write updates the Shadow register)

Bits	Field Name	Function	Initial Value
31:11	IntBaseAdd	Contains a pointer to the CPU 'intervention' area. The area includes 256 entries (2 32-bit words each) for the GT-48002's 'buffer_request' messages.	0x0
10:0	-	Reserved.	-

Device Table, Offset: 0x40

Bits	Field Name	Function	Initial Value
31:0	DevTab	GT-48002 Device Table. Each bit represents a GT-48002 device in the system. Bit 0 relates to device 0 etc.	0xffff

Time Out Counter, Offset: 0x4c

Bits	Field Name	Function	Initial Value
7:0	TimeOut0	Specifies in PCI clock units the number of clocks the GT-48002 holds the PCI bus before the generation of Retry termination. Used for the first data transfer.	0x0f (16 clocks)
15:8	TimeOut1	Specifies in PCI clock units the number of clocks the GT-48002 holds the PCI bus before the generation of Retry termination. Used for data transfers following the first data.	0x07 (8 clocks)
23:16	RetryCounter	Specifies the number of Retries	0xff

Port Counters (2 Blocks), Offset: 0x040000 - 0x0400c8

The CPU should read all the counters during initialization in order to reset the counters to '0'. All counters are 32-bit counters. The CPU is allowed to access the counters using single transactions (burst reads are not allowed).

Addr (for Port 0)	Counter Name	Function	Initial Value
0x040000	Bytes Received	The number of good bytes received. In countMode=1: Local and dropped packets increment the counter as well.	-
0x040004	Bytes Sent	The number of good bytes Sent	-
0x040008	Frames Received	The number of good frames received In countMode=1: Local and dropped packets increment the counter as well.	-
0x04000c	Frames Sent	The number of good frames sent	-
0x040010	Total Bytes Received	The number of bytes received (good and bad)	-
0x040014	Total Frames Received	The number of frames received (good and bad)	-
0x040018	Broadcast Frames Received	The l number of good broadcast frames received In countMode=1: Local and dropped broadcast packets increment the counter as well.	-
0x04001c	Multicast Frames Received	The l number of good multicast frames received In countMode=1: Local and dropped multicast packets increment the counter as well.	-
0x040020	CRC Error	The l number of frames with valid packet length and CRC error or misaligned (odd number of nibbles)	-
0x040024	Oversize Frames	The number of good frames with length more than 1518 bytes (or 1522 bytes for LongEn=1)	-
0x040028	Fragments	The number of good frames with length less than 64 bytes (not incliding colliding packets)	-
0x04002c	Jabber	The number of frames with length more than 1518 bytes (or 1522 bytes for LongEn=1) and with CRC error or misaligned (odd number of nibbles)	-

Addr (for Port 0)	Counter Name	Function	Initial Value
0x040030	Collision	The number of Collisions	-
0x040034	Late Collision	The number of Late Collisions (collisions occuring later than the 576th transmitted bit)	-
0x040038	Frames 64 Bytes	The number of good packets with length of 64 bytes	-
0x04003c	Frames 65-127 Bytes	The number of good packets with length between 65-127 bytes	-
0x040040	Frames 128-255 Bytes	The number of good packets with length between 128-255 bytes	-
0x040044	Frames 256-511 Bytes	The number of good packets with length between 256-511 bytes	-
0x040048	Frames 512-1023 Bytes	The number of good packets with length between 512-1023 bytes	-
0x04004c	Frames 1024-1522 Bytes	The number of good packets with length between 1024-1522 bytes	-
0x040050	MAC Rx Error	The number of frames with RX MAC Error (assertion of RxEr) .	-
0x040054	Dropped Frames	The number of dropped frames. Dropped packets are those which a receive buffer could not be allocated.	-

PCI Global Counters, Offset: 0x140040 - 0x140044

The CPU should read all the counters during initialization in order to reset the counters to ‘0’. All counters are 32-bit counters.

Address	Counter Name	Function	Initial Value
0x140040	PCIFraRec	Good Frames Received from the PCI	--
0x140044	PCIFraSent	Good Frames Sent to the PCI	--

SMI Register, Offset: 0x14004c

Bits	Field Name	Function	Initial Value
15:0	Data	**For SMI READ operation:** Two PCI transactiona are required: (1) PCI write to the SMI register with OpCode = 1, PhyAd, RegAd with the Data being any value. (2) PCI read from the SMI register. When reading back the SMI register, the Data is the addressed Phy register contents if the ReadValid bit (#27) is 1. The Data remains undefined as long as ReadValid is 0. **For SMI WRITE operation:** One PCI transaction is required: PCI write to the SMI register with OpCode = 0, PhyAd, RegAd with the Data to be written to the addressed Phy register.	N/A
20:16	PhyAd	Phy device address	0x0
25:21	RegAd	Phy device register address	0x0

Bits	Field Name	Function	Initial Value
26	OpCode	0 - Write 1 - Read	0x1
27	ReadValid	1 - indicates that the Read operation has been completed for the addressed RegAd register, and the data is valid on the Data field.	0x0
31:28	N/A	This bits should be driven 0x0 during any write to the SMI register.	0x0

3.4 PCI Configuration

The GT- 48001 contains the required PCI configuration registers. These registers are accessed from the PCI

Device and Vendor ID, Offset: 0x000

Bits	Field Name	Function	Initial Value
15:0	VenId	Provides the manufacturer of the GT-48002 (0x11ab).	0x11ab
31:16	DevId	Provides the unique GT- 48002 ID number .	0x4802

Status and Command, Offset: 0x004

Bits	Field Name	Function	Initial Value
1	MemEn	Memory Enable. Controls the GT-48002's response to memory accesses, as found in the PCI specification. 0 - Disable 1 - Enable	0x1
2	MasEn	Master Enable. Controls the GT-48002's ability to act as a master on the PCI bus. 0 - Disable 1 - Enable	0x1
4	MemWrInv	Memory Write and Invalidate. Controls the GT-48002's ability to generate Memory Write and Invalidate commands on the PCI bus. 0 - Disable 1 - Enable	0x0
5	Reserve	Reserved. Read only.	0x0
6	ParEn	Parity Enable. Controls the GT-48002's ability to respond to parity errors on the PCI. 0 - Disable 1 - Enable	0x0

Bits	Field Name	Function	Initial Value
8	SysErrEn	System Error Enable. Controls the GT-48002's ability to assert the SErr* pin. 0 - Disable 1 - Enable	0x0
22:9	Reserve	Reserved. Read only.	0x0
23	TarFastBB	Target Fast Back-to-Back. This indicates that the GT-48002 is capable of accepting Fast Back-to-Back transactions on the PCI bus. Read-only bit.	0x1
24	DataParDet	Data Parity Detected. This bit is set by the GT-48002 when it detects a Data Parity Error during a master operation. This bit is cleared by writing '1' to it.	0x0
26:25	DevSelTim	Device Select Timing. These bits indicate the GT-48002's DevSel* timing. The GT-48002's DevSel* timing is always set at medium (01), as defined in the PCI specification. Read only.	0x1
28	TarAbort	Target Abort. This bit is set upon Target Abort. This bit is cleared by writing '1' to it.	0x0
29	MastAbort	Master Abort. This pin is set upon Master Abort. This bit is cleared by writing '1' to it.	0x0
30	SysError	System Error. This pin is set upon System Error. This bit is cleared by writing '1' to it.	0x0
31	DetParErr	Detected Parity Error. This pin is set upon detection of Parity Error (in both master and slave operations). This bit is cleared by writing '1' to it.	0x0

Class Code and Revision ID, Offset: 0x008 (Read-only register)

Bits	Field Name	Function	Initial Value
7:0	RevID	Revision ID. Indicates the GT-48002 revision number.	0x0
23:16	SubClass	SubClass. Indicates the GT-48002 Subclass (0x0 - Ethernet).	0x0
31:24	BaseClass	Base Class. Indicates the GT-48002 Base Class (0x2 - Network Device) .	0x2

Header Type, Latency Timer, Cache Line, Offset: 0x00c

Bits	Field Name	Function	Initial Value
7:0	CacheLine	Cache Line. Specifies the GT-48002's cache line size (size=8). Read only.	0x7
15:8	LatTimer	Latency Timer. Specifies in units of PCI bus clocks the value of the latency timer of the GT-48002. Default is 256 cycles (0xff).	0xff
23:16	HeadType	Header Type. Specifies the layout of bytes 10 hex through 3f hex. -	0x0

For more information on these fields, please refer to the PCI specification.

DRAM Base Address, Offset: 0x010

Bits	Field Name	Function	Initial Value
20:0	(Clear)	These bits are cleared.	0x0
21	(Set)	This bit is set (DRAM being addressed).	0x1
26:22	DevNum	Device Number. These bits specify the Device Number	DAddr[4:0] at reset
31:27	BaseAdd	Base Address. DRAM base address.	0x1

Internal Registers Base Address, Offset: 0x014

Bits	Field Name	Function	Initial Value
20:0	(Clear)	These bits are cleared.	0x0
21	(Clear)	This bit is clear (Internal Registers being addressed).	0x0
26:22	DevNum	Device Number. These bits specify the Device Number.	DAddr[4:0] at reset
31:27	BaseAdd	Base Address. DRAM base address.	0x1

Interrupt Pin and Line, Offset: 0x03c

Bits	Field Name	Function	Initial Value
7:0	IntLine	Interrupt Line. Provides interrupt line routing information.	0x0
15:8	IntPin	Interrupt Pin. Indicates which interrupt pin the GT-48002 uses. The GT-48002 uses INTA in the PCI slot.	0x1

4 PINOUT TABLE

4.1 208 pin PQFP (sorted by number)

Pin #	Signal Name	Pin #	Signal Name	Pin #	Signal Name
1	VSS	36	DData[5]	71	RxD0[2]
2	VDD	37	DData[4]	72	RxD0[1]
3	AD[4]	38	DData[3]	73	RxD0[0]
4	AD[3]	39	DData[2]	74	RxEr[0]
5	AD[2]	40	DData[1]	75	VDD
6	AD[1]	41	DData[0]	76	RxClk[0]
7	AD[0]	42	VDD	77	VSS
8	DData[31]	43	VSS	78	RxDV[0]
9	DData[30]	44	CAS*	79	CrS[0]
10	DData[29]	45	WE*	80	Col[0]
11	DData[28]	46	RAS[1]*	81	TxEn[1]
12	DData[27]	47	RAS[0]*	82	VDD
13	DData[26]	48	ChipSel*	83	TxClk[1]
14	DData[25]	49	DAddr[8]	84	VSS
15	DData[24]	50	DAddr[7]	85	TxD1[3]
16	DData[23]	51	DAddr[6]	86	TxD1[2]
17	DData[22]	52	DAddr[5]	87	TxD1[1]
18	DData[21]	53	DAddr[4]	88	TxD1[0]
19	DData[20]	54	DAddr[3]	89	VSS
20	DData[19]	55	DAddr[2]	90	VDD
21	DData[18]	56	DAddr[1]	91	VDD
22	DData[17]	57	DAddr[0]	92	VSS
23	DData[16]	58	VDD	93	RxD1[3]
24	VDD	59	VSS	94	RxD1[2]
25	VSS	60	TxEn[0]	95	RxD1[1]
26	DData[15]	61	VDD	96	RxD1[0]
27	DData[14]	62	TxClk[0]	97	RxEr[1]
28	DData[13]	63	VSS	98	VSS
29	DData[12]	64	TxD0[3]	99	VDD
30	DData[11]	65	TxD0[2]	100	RxClk[1]
31	DData[10]	66	TxD0[1]	101	VSS
32	DData[9]	67	TxD0[0]	102	RxDV[1]
33	DData[8]	68	VSS	103	CrS[1]
34	DData[7]	69	VDD	104	Col[1]
35	DData[6]	70	RxD0[3]	105	VDD

Pin #	Signal Name	Pin #	Signal Name	Pin #	Signal Name
106	VSS	141	VSS	176	AD[17]
107	MDC	142	VDD	177	AD[16]
108	MDIO	143	VSS	178	CBE[2]*
109	No Connect	144	Int*	179	Frame*
110	MAC1LED[5]	145	Rst*	180	VSS
111	MAC1LED[4]	146	VDD	181	VDD
112	MAC1LED[3]	147	Clk	182	IRdy*
113	MAC1LED[2]	148	VSS	183	TRdy*
114	MAC1LED[1]	149	Gnt*	184	DevSel*
115	MAC1LED[0]	150	Req*	185	Stop*
116	ForceLinkPass*	151	VSS	186	PErr*
117	MAC0LED[5]	152	VDD	187	VSS
118	MAC0LED[4]	153	AD[31]	188	VDD
119	MAC0LED[3]	154	AD[30]	189	SErr*
120	MAC0LED[2]	155	AD[29]	190	Par
121	MAC0LED[1]	156	AD[28]	191	CBE[1]*
122	MAC0LED[0]	157	VSS	192	AD[15]
123	VDD	158	VDD	193	AD[14]
124	VSS	159	AD[27]	194	VSS
125	EnDev*	160	AD[26]	195	VDD
126	RstQueue*	161	AD[25]	196	VSS
127	VSS	162	AD[24]	197	AD[13]
128	VSS	163	CBE[3]*	198	AD[12]
129	VSS	164	IdSel	199	AD[11]
130	DisWD*	165	VSS	200	AD[10]
131	EnELScrubing*	166	VDD	201	AD[9]
132	SkipInit*	167	AD[23]	202	VSS
133	Scan*	168	AD[22]	203	VDD
134	TriState*	169	AD[21]	204	AD[8]
135	VSS	170	VSS	205	CBE[0]*
136	LEDClk	171	VDD	206	AD[7]
137	LEDData	172	AD[20]	207	AD[6]
138	LEDStb	173	AD[19]	208	AD[5]
139	LEDMode	174	VSS		
140	VDD	175	AD[18]		

5 DC CHARACTERISTICS - *PRELIMINARY/SUBJECT TO CHANG E*

5.1 Absolute Maximum Ratings

Symbol	Parameter	Min.	Max.	Unit
Vdd	Supply Voltage	-0.3	6.5	V
Vi	Input Voltage	-0.3	Vdd+0.3	V
Vo	Output Voltage	-0.3	Vdd+0.3	V
Io	Output Current		24	mA
Iik	Input Protect Diode Current		+-20	mA
Iok	Output Protect Diode Current		+-20	mA
Top	Operating Temperature	0	85	^{o}C
Tstg	Storage Temperature	-40	125	^{o}C
ESD			2000	V
Pt	Power Dissipation		1.5	W

5.2 Recommended Operating Conditions

Symbol	Parameter	Min.	Typ.	Max.	Unit
Vdd	Supply Voltage	4.75		5.25	V
Vi	Input Voltage	0		Vdd	V
Vo	Output Voltage	0		Vdd	V
Top	Operating Temperature	0		70	^{o}C
Cin	Input Capacitance		7.2		pF
Cout	Output Capacitance		7.2		pF

5.3 DC Electrical Characteristics Over Operating Range

(TC=0-70^{o}C; Vdd=+5V, +/-5%)

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Unit
Vih	Input HIGH level	Guaranteed Logic HIGH level	2.0		Vdd + 0.5V	V
Vil	Input LOW level	Guaranteed Logic LOW level	-0.5		0.8	V
Voh	Output HIGH Voltage	IoH = 2 mA IoH = 4 mA IoH = 8 mA IoH = 12 mA IoH = 16 mA IoH = 24 mA	2.4			V

Symbol	Parameter	Test Condition	Min.	Typ.	Max.	Unit
Vol	Output LOW Voltage	IoL = 2 mA IoL = 4 mA IoL = 8 mA IoL = 12 mA IoL = 16 mA IoL = 24 mA			0.4	V
Iih	Input HIGH Current				+-1	uA
Iil	Input LOW Current				+-1	uA
Iozh	High Impedance Output Current				+-1	uA
Iozl	High Impedance Output Current				+-1	uA
Vh	Input Hysteresis	Vdd = 4.5V Vdd = 5.0V Vdd = 5.5V	0.52 0.54 0.56		0.60 0.61 0.62	mV
Icc	Operating Current				220	mA

NOTE:

Pullup/Pulldown resistors are 45KOhm minimum, 65KOhm typical, 80KOhm maximum.

6 AC TIMING - *PRELIMINARY/SUBJECT TO CHANG E*

(TC= 0-70^{0}C; VDD= +5V, +/- 5%)

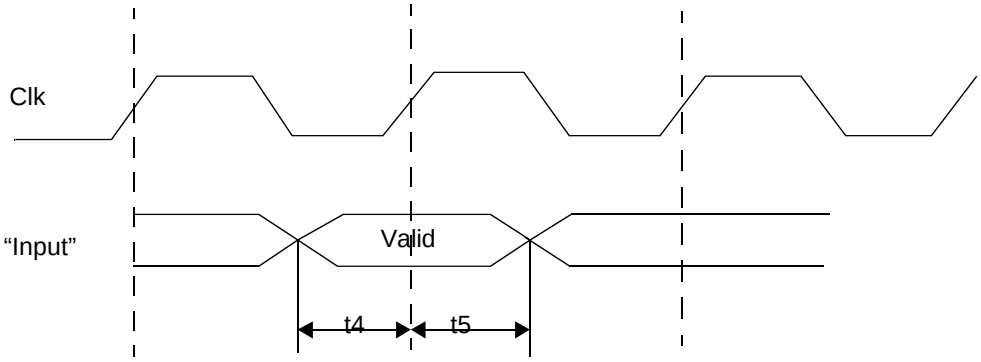
Symbol	Signals	Description	Min	Max	Unit
	Clk, Rst*, Frame*, IRdy*, TRdy*, DevSel*, Stop*, PErr*, Par, Int* AD[31:0], CBE[3:0]*, Gnt*, IdSel, Req*, SErr*	See PCI Specification Rev. 2.1.			
t3	DAddr[8:0], DData[31:0], CAS*, RAS*, WE*, ChipSel*	Delay from Clock Rising or Falling Edge	2	17	nS
t4	DData[31:0], Rst-Queue*, EnDev*	Setup	10		nS
t5	DData[31:0], Rst-Queue*, EnDev*	Hold	1		nS
t6	DData[31:0]	Float Delay	2	18	nS
t7	DData[31:0]	Drive Delay	2	12	nS

Notes:

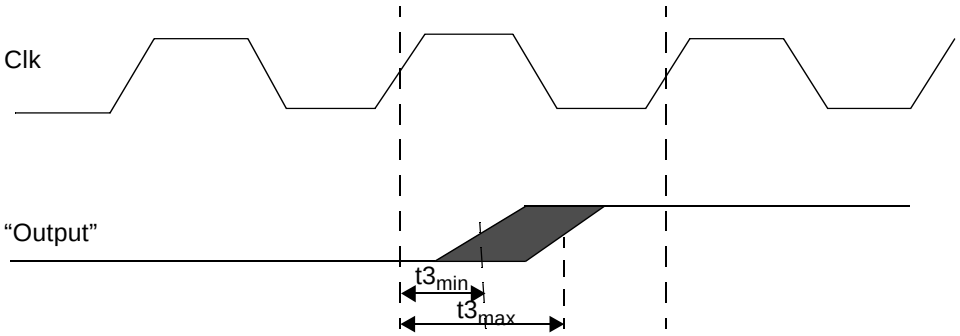
1. All Delays, Setup, and Hold times are referred to Clk rising edge, unless stated otherwise.

2. All outputs are specified for 50pf load.

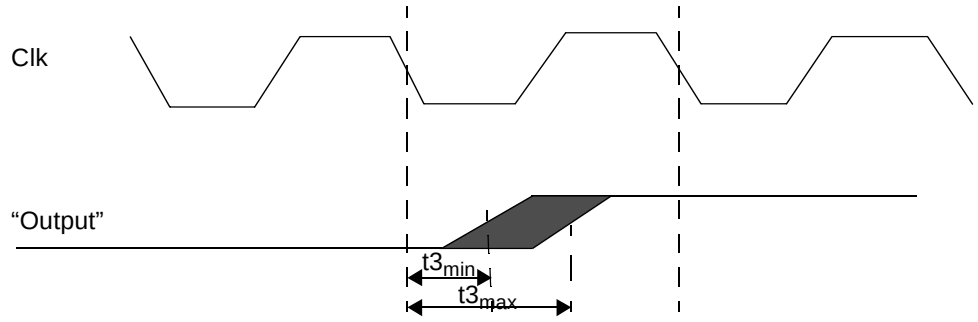
3. “All Inputs” and “All Outputs” also refer to I/O signals’ behavior.



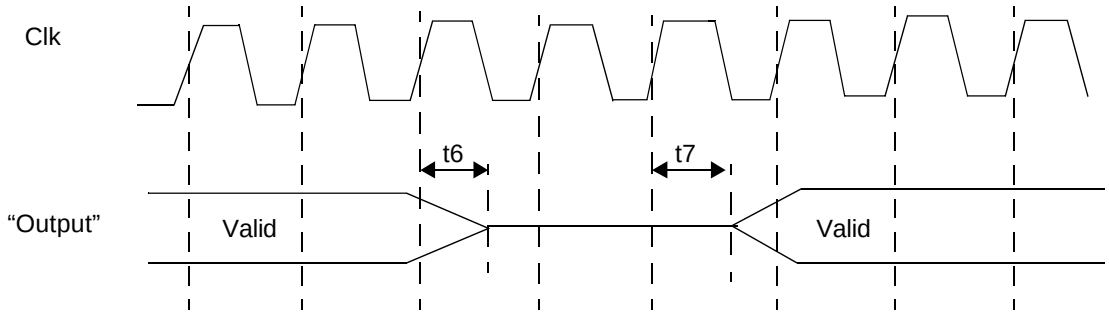
Inputs Setup and Hold



Output Delay from Clock Rising Edge



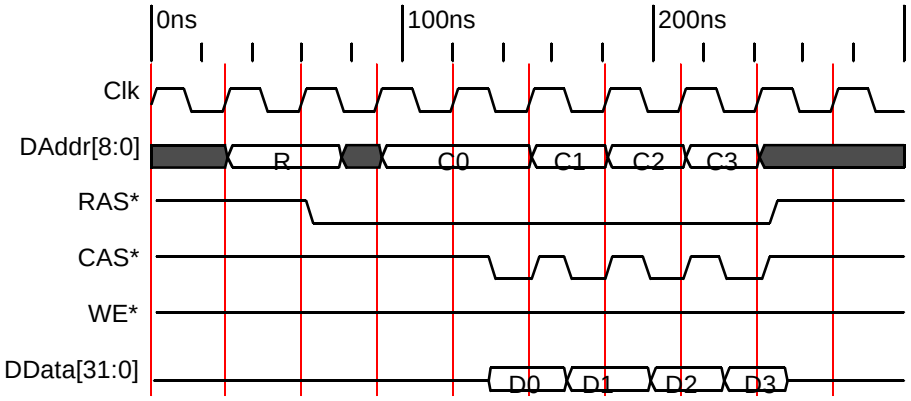
Output Delay from Clock Falling Edge



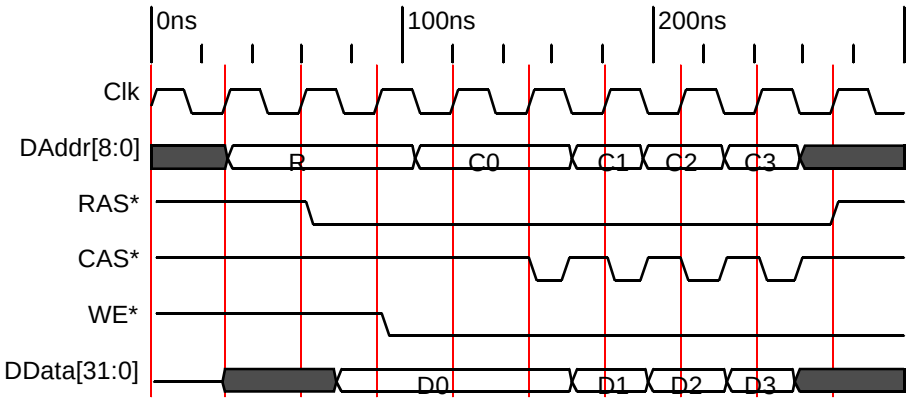
Output Float and Drive Delay

7 FUNCTIONAL WAVEFORMS

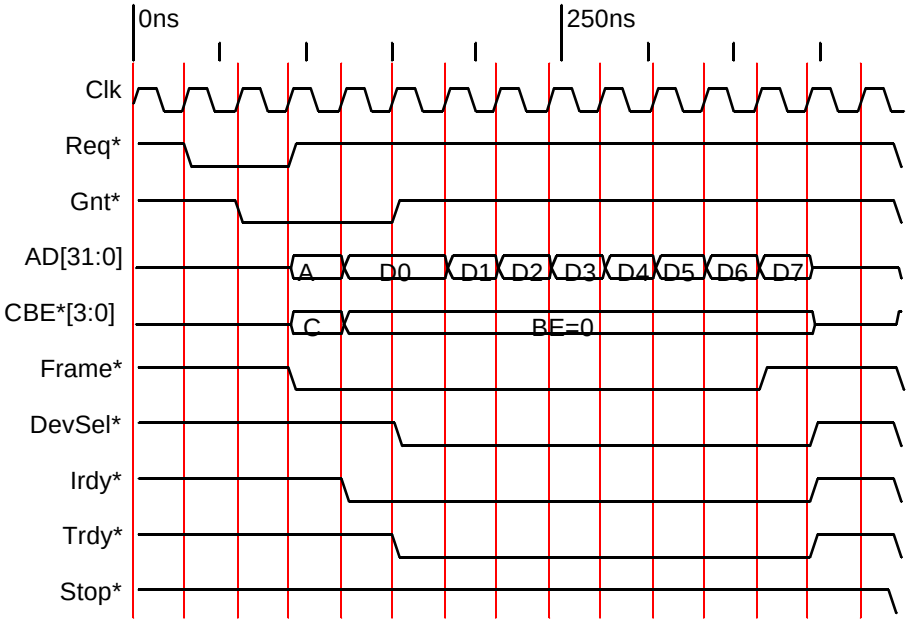
7.1 EDO Read



7.2 EDO Write



7.3 PCI Read/Write Cycle



8 SYSTEM CONFIGURATIONS

8.1 Switch Expansion

Figure 6 describes the way to expand the switch. Multiple GT-48001 and GT-48002 devices are connected to a local PCI bus (up to 4. The PCI buses are connected via PCI-to-PCI bridges. These devices are being used to filter the traffic between multiple PCI buses, and therefore to increase the effective aggregate bandwidth in the switch. The local traffic between GT-48002 devices in one PCI bus is not forwarded to the other GT-48002 devices. Only the traffic between GT-48002 devices in different PCI buses crosses the bridges.

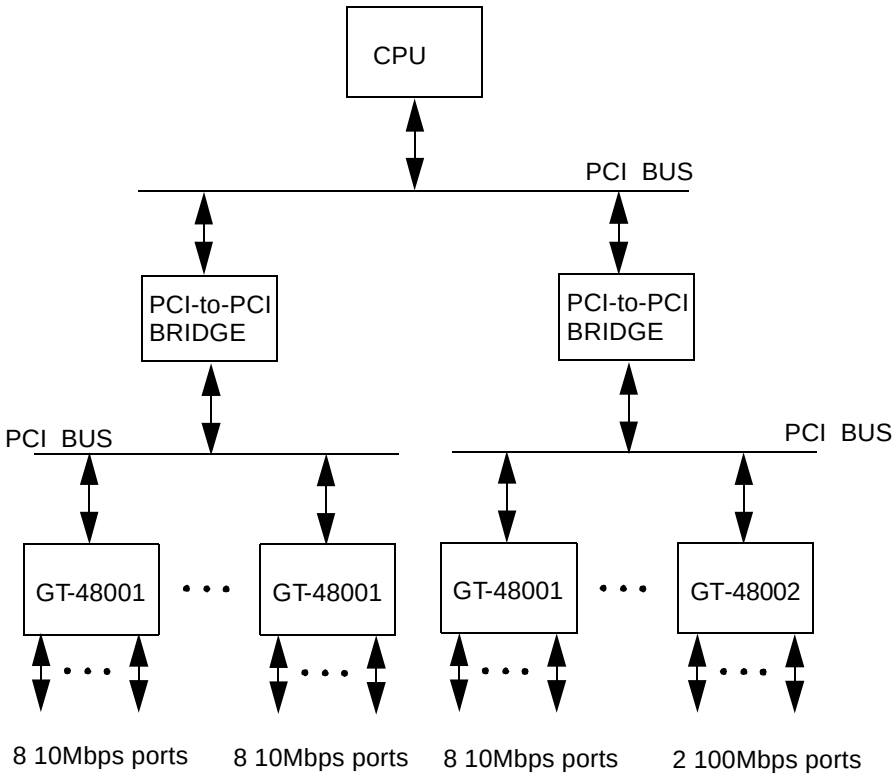


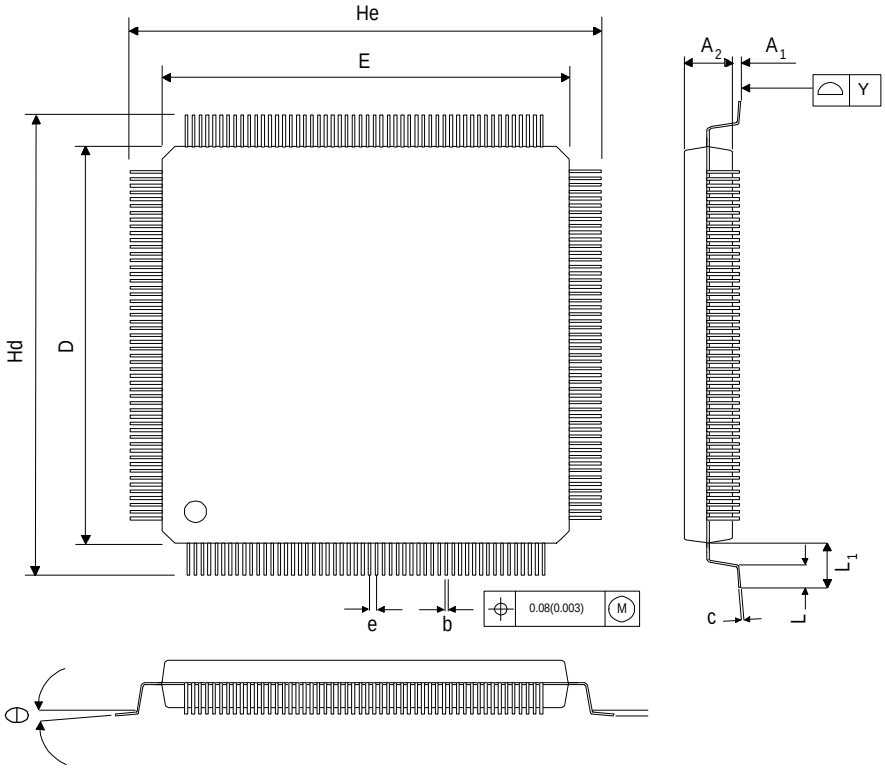
Figure 6 - Switch Expansion

9 Revision History

Table 1: Revision History

Document Type	Revision Number	Date	Comments
PRELIMINARY REV.	1.0	9/96	First revision of PRELIMINARY REVISION for general distribution.
PRELIMINARY REV.	1.1	10/96	Pinout Correction (Swap 109 and 116) . Revision History Table Added .

10 PACKAGING



208 LEAD PQFP PACKAGE OUTLINE

SYMBOL	MILLIMETERS		
	MIN.	NOM.	MAX.
A_1	0.05	0.25	0.50
A_2	3.17	3.32	3.47
b	0.10	0.20	0.30
c	0.10	0.15	0.20
D	27.90	28.00	28.10
E	27.90	28.00	28.10
e		0.50	
Hd	30.35	30.60	30.85
He	30.35	30.60	30.85
L	0.45	0.60	0.75
L_1		1.30	
Y			0.08
Θ	0		7

http://www.galileoT.com
info@galileoT.com

Galileo Technology™

1735 N. First St. #308
San Jose, CA 95112
USA
Tel (408) 451-1400
Fax (408) 451-1404