Galileo Technology™

Switched Ethernet Controller

GT- 48001

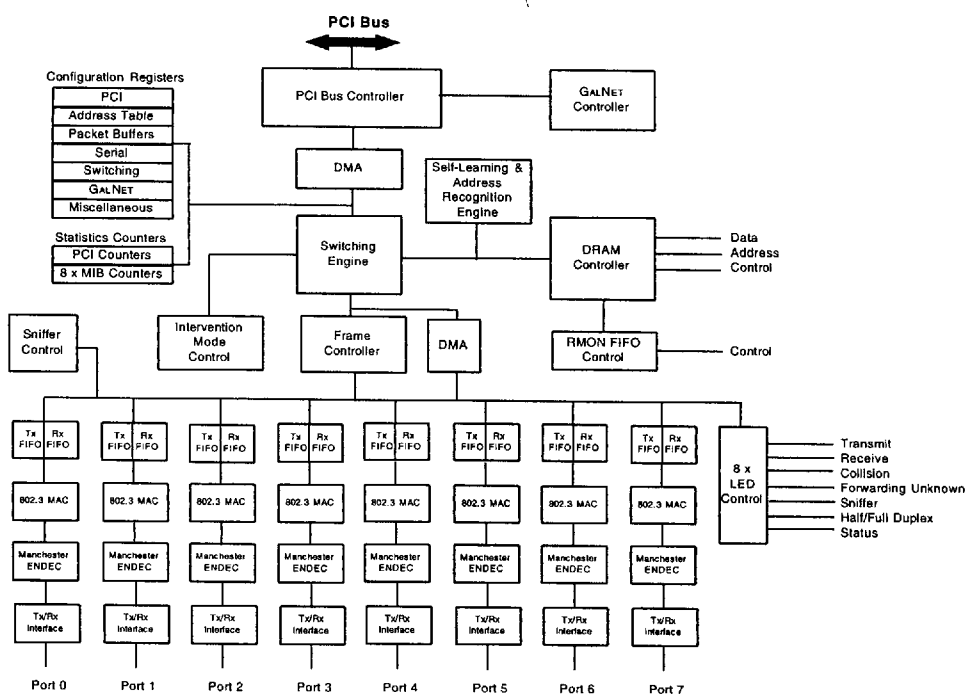
Preliminary Rev. 1.1
December 19, 1996

FEATURES

A Switched Ethernet Controller FAQ is located in the library section of Galileo's www site (http://www.galileoT.com).

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

- First product using the GalNet™ architecture
- High integration, low cost, Switched Ethernet Controller
- Provides switching functions between 8 Ethernet ports
- Enables switch expansion via a high performance, high bandwidth PCI bus (up to 1Gbps)
- Supports 'Store and Forward' switching approach
- Low last-bit to first-bit delay
- Provides aggregate throughput of more than 650K Unicast packets/sec and more than 90K Multicast packets/sec
- Permits forwarding and filtering at full wire speed of 14,880 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 eight 802.3 compliant Ethernet ports
  - Dedicated 10Mbps or full-duplex 20Mbps Ethernet bandwidth for each port
  - Serial mode selectable per port
  - Modes are 10Base-T, 10Base-FL, AUI, and NRZ Synchronous
- All digital logic on-chip for each port
  - Media Access Control (MAC)
  - Manchester encoder/decoder
  - Link integrity, Partition
  - Automatic polarity detection and correction
  - Dual 32-byte FIFOs for receive and transmit
  - 7 LEDs for Link Status, Receive, Transmit, Collision, Forward Unknown Packets, Port Sniffer, and Half/Full Duplex.
- Direct support for packet buffering
  - Interfaces directly to 1Mbyte or 2Mbyte 32-bit, 60ns standard or 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 6 GT-48001 devices per PCI slot without PCI-to-PCI bridging, and up to 32 GT-48001 devices (256 ports) in a switch
  - CPU connection for management
  - Connection to high speed network
- Various management support features
  - Repeater MIB and PCI counters
  - Aging support
  - Hardware assist for Spanning Tree algorithm
  - Station-to-Station connectivity matrix (for RMON)
  - CPU access to Address Table
  - Ability to define static addresses
  - Monitoring (sniffer) mode
- 30MHz-33MHz clock rate
- 5V operation



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

- 208 pin QFP package

OVERVIEW

General Description

The GT-48001 is a high performance, low cost, Switched Ethernet Controller. It provides the switching functions between eight dedicated 10Mbps/20Mbps Ethernet ports. Switch expansion of up to 256 ports is enabled via a high performance, high bandwidth PCI bus.

The GT-48001 is the first member of Galileo's GalNet switching architecture. It enables designers to build an intelligent and scalable switching hub and lets the user boost performance cost-effectively in departments, work-groups and small data centers.

The GT-48001 uses a Store and Forward switching approach. It forwards and filters at more than full wire speed. The switching engine provides aggregate throughput of 650K Unicast packets/sec and 90K Multicast packets/sec.

Serial Interface

The GT-48001 incorporates eight Ethernet ports. Each port works at 10Mbps or 20Mbps (full-duplex) and includes the Media Access Control (MAC), Manchester Encoder/Decoder, Link Integrity logic, Auto Polarity and seven LED controls.

The GT-48001 is compliant with both 802.3 and Ethernet specifications. It supports different PHY media such as 10Base-T, 10Base-5, 10Base-2 and 10Base-FL.

Address Recognition

The GT-48001 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-48001 provides a self address learning mechanism. Each device holds its own Address Table. The GT-48001 learns the new addresses as they arrive from the wire and updates all the Address Tables in the system.

DRAM Interface

The GT-48001 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 standard page mode DRAMs and EDO DRAMs.

The GT-48001 supports up to 1024 (full packet size) receive buffers. These buffers are dynamically allocated to the eight receive ports and the PCI side. The number of buffers can be optionally limited per port basis.

PCI Interface

The GT-48001 has a glueless connection to the PCI bus. It is compliant with PCI Rev 2.1. The GT-48001 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 and to increase the number of ports. Up to six GT-48001 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 (256 ports).

The PCI bus is also used to connect a CPU for management / routing functions and to connect a high speed LAN adapter such as ATM or Fast-Ethernet. The connection to a CPU is optional.

Management Features

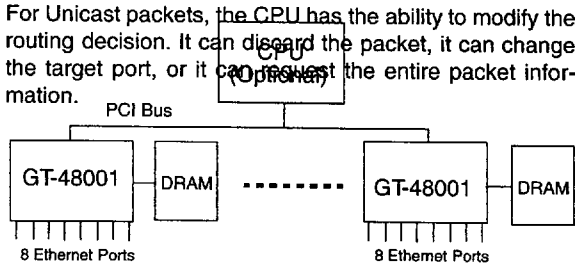
The GT-48001 provides comprehensive management capabilities enabling the user to manage and balance the network.

The GT-48001 supports per-port statistic 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-48001 also provides a Station-to-Station Connectivity matrix and the ability to select a port to work in Monitoring (Sniffer) mode.

Intervention mode

The GT-48001 incorporates an enhanced feature called 'Intervention' mode. This feature permits software or hardware intervention in the packet routing decision mechanisms. The support is done 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-48001 devices.

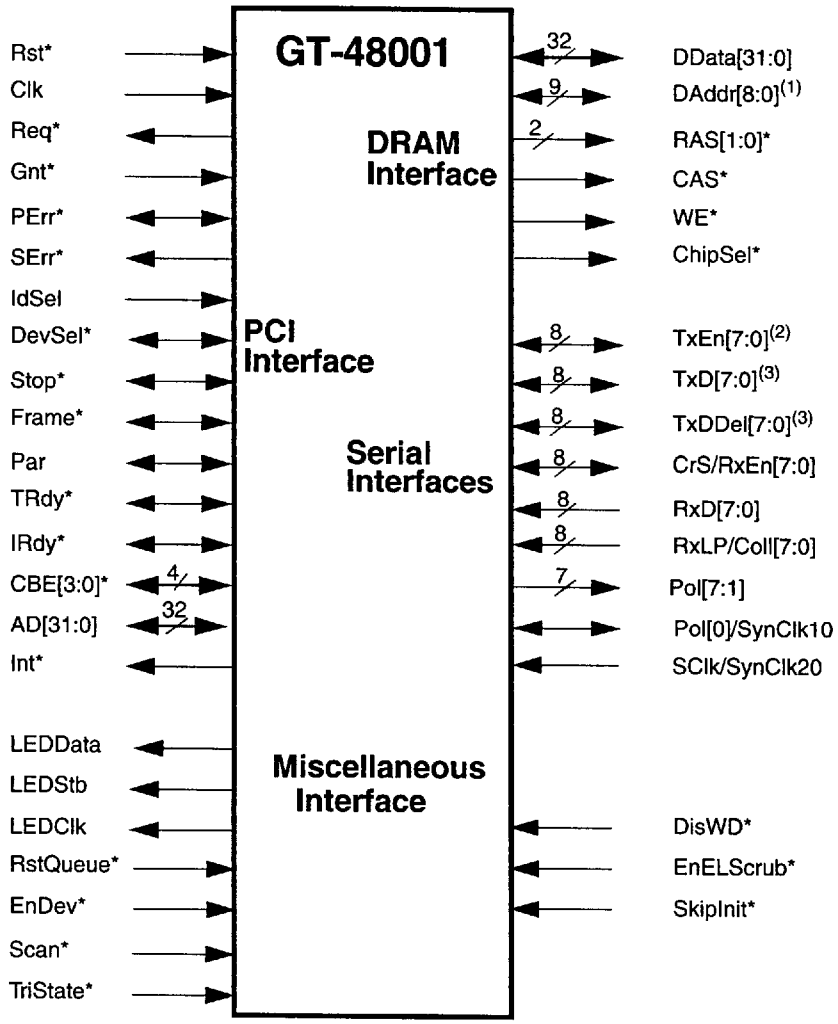
For Unicast packets, the CPU has the ability to modify the routing decision. It can discard the packet, it can change the target port, or it can request the entire packet information.



Easy Expansion To 256-Ports and CPU Connectivity Via Familiar PCI Bus

1 PIN INFORMATION

1.1 Logic Symbol



(1) - Pins sampled at reset to establish Device Number and DRAM Parameters
(2) - Pins sampled at reset to establish Half/Full Duplex Mode per port
(3) - Pins sampled at reset to establish Serial Mode operation per port

1.2 Pin Assignment Table

Symbol	Type	Description
PCI Interface		
Rst*	I	**Reset:** Resets the GT-48001 to its initial state. This signal must be asserted for at least 10 PCI clock cycles. When in the reset state, all PCI output pins are put into tristate and all open drain signals are floated.
Clk	I	**Clock:** Provides the timing for the GT-48001 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 range is 30-33MHz
Req*	O	**Bus Request:** Asserted to indicate to the bus arbiter that it desires use of the bus.
Gnt*	I	**Bus Grant:** Indicates to the GT-48001 that access to the bus is granted.
PErr*	I/O	**Parity Error:** Asserts when a data parity error is detected.
SErr*	O	**System Error:** Asserted when an address parity error is detected. The GT-48001 asserts the SErr* two cycles after the failing address.
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-48001 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-48001 aborts the cycle. As a target, when the GT-48001 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-48001 responds to the assertion of Stop*, either disconnecting, retrying, or aborting. As a target, the GT-48001 asserts Stop* to retry.
Frame*	I/O	**Cycle Frame:** Asserted by the GT-48001 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-48001 acts as a target, to detect a configuration or memory transaction.
Par	I/O	**Parity:** Calculated by the GT-48001 as an even parity bit for the AD[31:0] and CBE[3:0]* lines.
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.

Symbol	Type	Description
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 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 Byte Enables, which determine which bytes carry valid data.
Int*	O	**Interrupt Request Line:** Int* is asserted by the GT-48001 when one (or more) of the bits in the Interrupt Cause register is set.
DRAM Interface		
DData[31:0]	I/O	**DRAM Data:** 32-bit Standard or 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-48001 to indicate the Device Number and the DRAM Parameters (see Reset Configuration section). Values are determined by connecting the appropriate pull-up/pull-down resistors. The Device Number and the DRAM Size are read by the CPU from the Status register.
RAS[1:0]*	O	**Row Address Strobes:** DRAM row address strobes. RAS[0]* is used for Bank0. RAS[1]* is used for Bank1.
CAS*	O	**Column Address Strobes:** DRAM column address strobe.
WE*	O	**Write Enable:** DRAM write enable.
ChipSel*	O	**Chip Select:** This pin is connected to an external FIFO device. It is asserted by the GT-48001 when the packet's Byte Count, Destination Address and Source Address are read from the DRAM. This information is stored in the FIFO and accessed by an external CPU for Station-to-Station connectivity matrix implementation.
Serial Interfaces		
TxEn[7:0]	I/O	**Transmit Enable:** In normal operation, it envelopes the transmitted packet. During reset, this multiplexed input indicates the port mode operation (pull-up for full duplex, pull-down for half duplex, see Reset Configuration section).
TxD[7:0]	I/O	**Transmit Data:** In normal operation, it outputs the Transmit Data. During reset, this multiplexed input pin behaves as SerMode0, which together with TxDDel (SerMode1) indicate the port mode of operation based on pull-up/pull-down resistors connected to them (see Reset Configuration section).
TxDDel[7:0]	I/O	**Transmit Data Delayed:** In normal operation, it outputs the Transmit Data Delayed. TxDDel follows the TxD signal by 50nsec. During reset, this multiplexed input pin behaves as SerMode1, which together with TxD (SerMode0) indicate the port mode of operation based on pull-up/pull-down resistors connected to them (see Reset Configuration section).
CrS/RxEn[7:0]	I/O	**Carrier Sense/Receive Enable:** In 10Base-T, AUI and 10Base-FL, this output pin indicates the Carrier Sense. In Sync mode, this input pin envelopes the receive packet.
RxD[7:0]	I	**Receive Data**

Symbol	Type	Description
RxLP/Coll[7:0]	I	**Receive Link Pulses/Collision:** This multiplexed pin carries the Receive Link Pulses in 10Base-T. In AUI and Sync mode, it indicates Collision.
Pol[7:1]	O	**Polarity:** Indicates the line polarity of Ports 7:1 in 10Base-T mode.
Pol[0]/SynClk10	I/O	**Polarity/Synchronous Clock 10:** Indicates the polarity of Port 0 in 10Base-T mode. In Synchronous mode, this input pin carries the 10MHz synchronous clock.
SClk/SynClk20	I	**80MHz Serial Clock/Synchronous Clock 20:** Used to recover the receive clock and to generate the 10MHz transmit clock. In Synchronous mode, this pin carries the 20MHz synchronous clock.
Misc. Interface		
LEDData	O	**LED Data:** LED indicators (Link Status, Receive, Transmit, Collision, Unknown, Port Sniffer, and Half/Full Duplex) of each port. The data is shifted out using the LEDClk and LEDStb pins.
LEDStb	O	**LED Strobe:** Indicates the beginning of valid data on the LEDData pin.
LEDClk	O	**LED Clock:** This pin is used to clock the serial data out.
RstQueue*	I	**Reset Transmit Queues:** When asserted, all internal transmit and receive queues are cleared. All GT-48001 state machines are moved to their initial state.
EnDev*	I	**Enable Device:** Enables serial and PCI ports. When asserted, all serial ports and the PCI port are active. When deasserted, the ports and the PCI are disabled.
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-48001 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* indicate the GT-48001 mode of operation as follows: Scan* = 1, TriState* = 1 - Normal operation Scan* = 0, TriState* = 1 - Scan Mode (Not implemented yet) Scan* = 1, TriState* = 0 - The GT-48001 drives all outputs I/O pins to High impedance. Scan* = 0, TriState* = 0 - Internal SRAM test
TriState*	I	**Tri State:** This pin together with Scan* indicate the GT-48001 mode of operation as described above.

2 FUNCTIONAL DESCRIPTION

2.1 General Description

The GT-48001 provides the switching functions between 8 (or more) Ethernet ports. It has three interfaces:

a) Eight Ethernet 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-48001 receives the incoming packets from the wire, searches in the Address Table the Destination Address and forwards the packet to the appropriate port. The port can be either local or in a different GT-48001 device that resides on the same PCI bus. In parallel, the GT-48001 checks the Source Address and updates all the Address Tables in the system accordingly.

The GT-48001 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-48001 supports up to 8K different MAC addresses. The Address Table is located in the DRAM and is fully controlled by the GT-48001 (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.

	63	62	61	60	59	58 56	55 51	50 3	2	1	0
1	Is	Id	M	St	R	Port#	Dev#	Addr[0:47]	A	Sk	V
8K											

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-48001 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 eight ports in a GT-48001 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-48001 has a self-learning mechanism. It learns the Ethernet addresses in real time. The GT-48001 searches 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-48001 waits to the end of the packet (good CRC) and updates its Address Table. It also notifies the other

GT-48001 devices and the CPU by sending a 'new_address' message on the PCI. 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-48001 asserts Int* to notify the CPU that the Address Table was modified.

2. If the SA was found, the GT-48001 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 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-48001 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-48001 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 device via the PCI bus.

2. If the DA is a Unicast address and the address was not found (Unknown), the GT-48001 acts as if it is a Multicast packet and forwards it to the ports and the devices which were programmed to forward Unknown packets (bit 4 in the Command register).

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 the other 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-48001 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 (the Device Number written in the Table is equal to its number). This particular device updates the source GT-48001's Address Table with the new address (by in turn sending it a 'new_address' message).

2.3 GT-48001 Buffers and Queues

The GT-48001 incorporates nine transmit queues (for the 8 local ports and the PCI), and one common receive buffer. Figure 2 shows the GT-48001 queues. The receive buffers as well as the transmit queues are located in the DRAM. The GT-48001 includes the pointers to the transmit queues. The GT-48001 data structure components are the following:

1. Receive Buffer - A common Rx buffer for all ports. The buffer is divided into 390 or 1024 blocks (depends on the DRAM size) of 1.5KBytes (1536 bytes) each. The buffer contains the whole packet information.

2. Rx Empty List - A list of 390 or 1024 bits. Each bit contains the status of its appropriate receive block in the DRAM (empty or occupied).

3. Tx Descriptors - A set of 9 transmit descriptor rings. Each ring contains 512 descriptors. The descriptor size is 1 Long Word (32-bits) and contains the Block Address divided by 600hex (1.5K), the Byte Count and the Packet Type (Multicast or Unicast).

4. Read/Write Pointers - 9 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 devices, and between the CPU and a GT-48001.

2.4.1 Forwarding a Unicast Packet to a Local Port

The sequence is as follows:

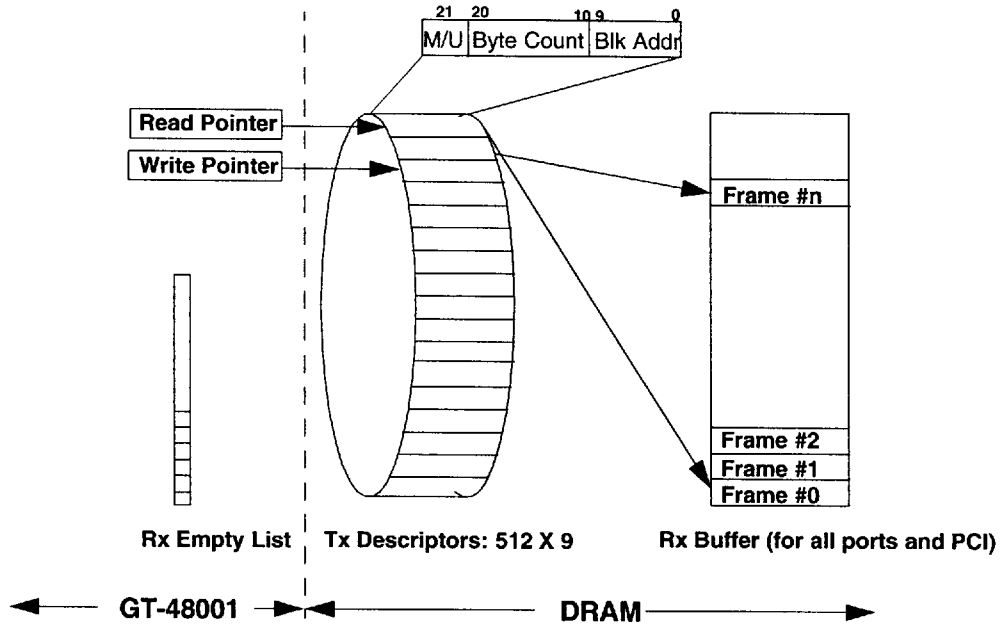


Figure 2 - GT-48001 Buffers and Queues

1. The incoming packet is fed to the Receive FIFO (there is an 8x32-bit FIFO per port) 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-48001 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-48001 transmits when the Write Pointer is not equal to the Read Pointer.

5. At the end of the packet transmit process, the GT-48001 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 4 32-bit bursts.

2. In parallel, an address recognition cycle is performed for both the DA and the SA. The GT-48001 uses the DA's Port Number and Device Number to queue the packet to the appropriate GT-48001 device and port.

3. At the end of a good packet transfer, the packet is entered into the PCI transmit queue (the 9th 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 clears 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-48001 transmits when the Write Pointer is not equal to the Read Pointer.

8. At the end of the packet transmit process, the GT-48001 increments the Read Pointer and clears the appropriate bit in the Empty List.

2.4.3 Forwarding a Multicast Packet

The GT-48001 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 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 all the GT-48001 devices.

2.4.4 Forwarding a Packet to the CPU

The GT-48001 forwards the packets directly to the CPU main memory. The GT-48001 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.

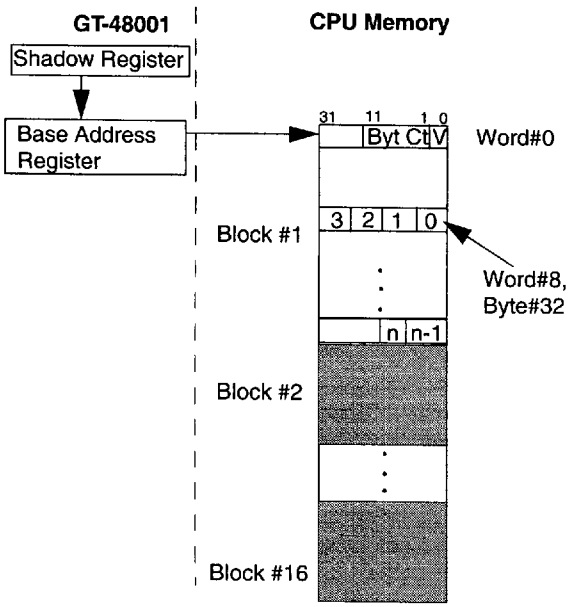


Figure 3 - CPU Data Structure

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]) and a Valid bit (bit 0). The Byte Count and the Valid bit are written at the end of the packet transfer. Words 2 to 7 are left empty for user purposes.

The communication between the GT-48001 and the CPU follows this sequence:

1. CPU updates the Base Address register

2. CPU updates the Shadow Register

3. GT-48001 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-48001 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 (33rd byte). Words 1 to 7 are left empty for user purposes.

4. At the end of the packet transfer, the GT-48001 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-48001 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-48001 holds a Status 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-48001 holds a Transmit Watchdog timer for each transmit queue. The default value of the timer is 60msec. The range is between 10 and 160msec. The timer measures the time between two consecutive packets which are being served. When the timer expires, the GT-48001 clears the appropriate used blocks and sends an interrupt to the CPU via Int*.

2.6 GT-48001 Device Table

The GT-48001 includes a 32-bit Device Table. Each bit in the table represents a different GT-48001 device in the system. Upon reset, each GT-48001 sets all the bits to '1'. The bits are cleared either by the CPU or upon PCI master abort (meaning that the GT-48001 tried to access a non-existent device). The Device Table is used by the GT-48001 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.

2.7 Intervention Mode

The GT-48001 supports a powerful mode called the Intervention Mode, which permits software or hardware intervention in the packet routing decision mechanisms. The support is done differently for Multicast and Unicast packets.

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-48001 device and each GT-48001 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-48001 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.

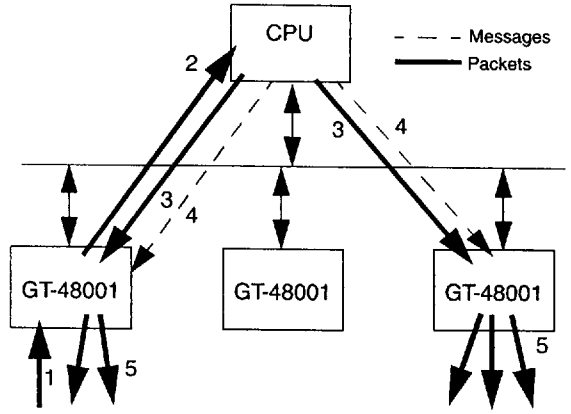


Figure 4 - Multicast Packet Transfer With Intervention

The sequence illustrated is as follows:

a. The Multicast packet is received and stored in the source GT-48001's DRAM (arrow #1).

b. The GT-48001 transfers the packet to the CPU main memory (arrow #2).

c. The CPU transfers the packet to the selected GT-48001 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-48001 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 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.

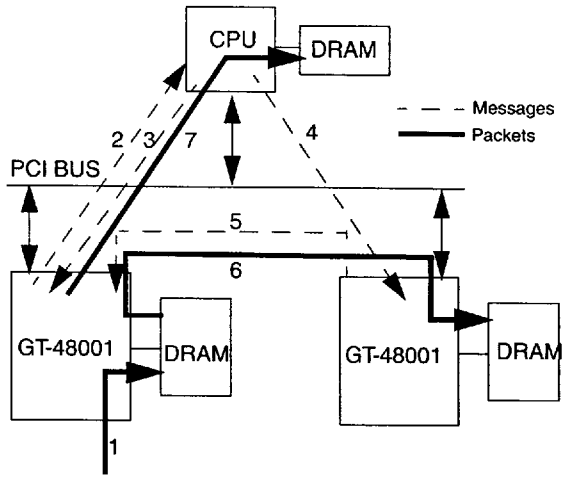


Figure 5 - Unicast Packet Transfer With Intervention

The sequence illustrated is as follows:

a. The Unicast packet is received and stored in the source GT-48001's memory (arrow #1).

b. If one of the Intervention bits is set, the GT-48001 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-48001 (arrow #5). The source GT-48001 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-48001 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-48001 supports the following management features:

- Repeater MIB and PCI counters
- Station-to-Station matrix
- Monitoring (Sniffer)

2.8.1 Repeater MIB and PCI Counters

The GT-48001 incorporates a full set of Repeater MIB counters per port, as well as PCI counters. The PCI counters are global for all 8 ports. The CPU periodically reads the counters. The Repeater MIB counters are the following:

- Bytes Received
- Multicast Bytes Received
- Broadcast Bytes Received
- Bytes Sent
- Frames Received
- Multicast Frames Received
- Broadcast Frames Received
- Frames Sent
- Collision
- Late Collision
- CRC + Alignment Error
- Jabber
- Frame Too Short
- Frame Too Long
- Bad (CRC Error, Frame Too Long) Bytes Received

The global PCI counters are:

- PCI Frames Received
- PCI Frames Sent

2.8.2 Station-to-Station Matrix

The GT-48001 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-48001. The GT-48001 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-48001 to work in Monitoring mode in one of the eight ports. This is done by setting bit 4 in the Port Control register. In this mode, the GT-48001 sends all receive (including local traffic) and transmit packets to the CPU or to a port in one of the GT-48001 devices which was assigned to be the Sniffer. The target sniffer is written into the CPU and Sniffer Numbers register.

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

2.8.4 Spanning Tree Support

The GT-48001 provides the required hardware assistance for Spanning Tree algorithm implementation. The Spanning Tree algorithm is performed by the CPU.

The GT-48001 holds a SpanEn bit in the Global Control register and additional SpanEn bits in each of the 8 Port Control registers. The following table summarizes the hardware assistance for the Spanning Tree algorithm.

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 CPU. All receive/transmit packets should be rejected. Accept BPDU) messages from the CPU. No address learning
1	0	Forward	Transfer BPDU to the CPU. Accept all packets. Address learning

Note: The GT-48001 does not learn during the traditional 'Learning' stage. It only starts learning during the Forward stage.

2.9 Serial Interfaces

The GT-48001 incorporates all the required digital circuitry to interface to 10Base-T, 10Base-5, 10Base-2, 10Base-FL, and NRZ Synchronous media. This includes the following:

2.9.1 Media Access Control (MAC)

The GT-48001 operates in half-duplex or full-duplex modes. In half-duplex mode, the GT-48001 checks that there is no competitor for the network bus before transmission. In addition to listening for a clear line before transmitting, the GT-48001 handles collisions in a predetermined way. If two nodes attempt to transmit at the same time, the signals collide and the data on the line is garbled. The GT-48001 listens while it is transmitting, and it can detect a collision. If a collision is detected, the GT-48001 transmits a 'JAM' pattern and then delays its retransmission for a random time period determined by the backoff algorithm. In full-duplex mode, the GT-48001 transmits unconditionally.

2.9.2 Manchester Encoder/Decoder

The Manchester Encoder receives clocked data from the transmit engine and uses an internal 20MHz clock (80MHz divided by 4) to provide the Manchester-encoded data to the Physical interface. The Manchester Decoder uses the 80MHz clock to recover the receive clock and to sample the incoming data.

2.9.3 Link Integrity and Auto Polarity Detector

The GT-48001 implements the Link Integrity test as specified in the IEEE 802.3 10Base-T and 10Base-FL supplements. In addition, the GT-48001 provides an Auto Polarity method for 10Base-T, to switch the polarity of the data going into the MAC layer accordingly.

2.9.4 LED Support

The GT-48001 supports seven LED Controls per port as follows:

- Transmit data in progress
- Receive data in progress
- Collision active
- Forwarding of Unknown packets enabled
- Port configured as Sniffer
- Half/Full Duplex
- Status

The LED control signals are serially shifted at 10MHz by the GT-48001 and latched by an external shift register. The LED Indications Interface consists of three outputs:

- LEDClk - primary clock
- LEDStb - start of data frame indication
- LEDData - multiplexed data output

2.10 PCI Interface

The GT-48001 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-48001 internal registers and addresses are memory mapped. Bit 21 in the address selects between the DRAM area and the internal registers. The GT-48001'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-48001 devices or between the CPU and a GT-48001 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]	Address Unknown/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; 29 - Port 7... 22 - Port 0)
[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; 29 - Port 7... 22 - Port 0)
[21]	Multicast/Unicast
[20:10]	Byte Count
[9:0]	Target Buffer Address (divided by 0x600)

2.11 DRAM Interface

The GT-48001 supports EDO DRAMs, as well as standard page mode DRAMs. The usable bandwidth in both modes satisfies the required bandwidth for data transfer, address recognition and Tx descriptor fetch/update.

EDO DRAM can be used to shorten the latency between packets.

Galileo recommends that all designs use EDO DRAM.

2.12 Reset Configuration

The GT-48001 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-48001 samples these input signals during reset.

Pin	Configuration Function
DAddr[4:0]	Device Number
DAddr[5]	DRAM Size
0-	2Mbyte
1-	1Mbyte
DAddr[8]	DRAM Type
0-	Standard
1-	EDO
TxEn/FDx[7:0]	Half/Full Duplex Mode Per Port
0-	Half Duplex
1-	Full Duplex
TxDDel[7:0],TxD[7:0]	Serial Mode Per Port (Note that the logic order is TxDDel/TxD)
00-	10Base-T
01-	10Base-FL
10-	AUI
11-	Sync

2.13 Unused PCI Bus in Standalone Systems

Some applications (such as stand alone mode) do not require use of the PCI Bus. The PCI bus pins must be connnected in following fashion for to insure proper operation of the GT-48001 when not using the PCI bus.

DevSel*	pulled up to VCC
Stop*	pulled up to VCC
Par*	pulled up to VCC
PErr*	pulled up to VCC
Frame	pulled up to VCC
IRdy*	pulled up to VCC
TRdy*	pulled up to VCC
Gnt*	pulled down to GND
IdSel*	pulled down to GND
SErr*	No Connect
Req*	No Connect
Int*	No Connect

All pull-up and pull-down resistors should have a value of 4.7KΩ.

3 REGISTER TABLES

The GT-48001 incorporates the required PCI Configuration registers, Command registers and various counters for management purposes. The GT-48001 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.

The actual address of an internal register is the sum of the GT-48001 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.

3.1 Register Map

Description	Offset
Internal	
Global Control	0x140028
Port Control 0	0x040200
Port Control 1	0x040204
Port Control 2	0x040208
Port Control 3	0x04020c
Port Control 4	0x040210
Port Control 5	0x040214
Port Control 6	0x040218
Port Control 7	0x04021c
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 - 0x040038
Port 1 Counter Block	0x040040 - 0x040088
Port 2 Counter Block	0x040080- 0x0400b8
Port 3 Counter Block	0x0400c0- 0x0400f8

Port 4 Counter Block	0x040100- 0x040138
Port 5 Counter Block	0x040140- 0x040178
Port 6 Counter Block	0x040180- 0x0401b8
Port 7 Counter Block	0x0401c0- 0x0401f8
PCI Global Counters	0x140040- 0x140044

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

3.2 Internal

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-48001 will not learn any new addresses from the wire	0x0
1	RMONEn	RMON Enable. 0 - RMON disabled 1 - The GT-48001 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-48001 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-48001 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

Bits	Field Name	Function	Initial Value
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 8 Ethernet ports. The 8 ports have equal priority. 1- PCI has higher priority than the 8 ports. The priority is: PCI, Port 0; PCI, Port 1;.....PCI, Port 7; PCI, Port 0.....	0x1
6	ForwUnk	Forward Unknown. It defines whether the GT-48001 will forward Unknown packets to the CPU or not. 0 - Do not forward 1 - Forward	0x0
7	ForwNewAdd	Forward New Address. It defines whether the GT-48001 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-48001 devices which was assigned to be the Sniffer. 0 - CPU type 1 - GT-48001 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-48001 reads the DA/SA of the packet that is forwarded only to the local ports. 1 - The GT-48001 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-48001 forwards Multicast packets to all the ports. 1 - Multicast packets forwarded only to the CPU.	0x0

Bits	Field Name	Function	Initial Value
23	ParEn	Partition Enable. When more than 32 collisions occur while transmitting, the GT-48001 enters the Partition mode. It waits for the first good packet from the wire, and then goes back to Normal mode. Under Partition mode it continues transmitting, but not receiving. 0 - Normal mode 1- Partition mode	0x0
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-48001 forwards BPDU packets only to the CPU.	0x0

Port Control (8 Registers), Offset: 0x040200 - 0x04021c

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	TxEn (at reset)
3:2	SerMode	Serial Mode. These bits indicate the serial mode. The logic order at reset is TxDDel/TxD=[3:2]. 00 - 10Base-T 01 - 10Base-FL 10 - AUI 11 - Synchronous	TxDDel/TxD (at reset)
4	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	0x0
6:5	Reserved	Reserved.	0x0
7	AutoPol	Auto-Polarity Disable. Enable/disable Auto-Polarity detection function. 0 - Disable 1 - Enable	0x0
9:8	Reserved	Reserved.	0x2
10	FilBroad	Filter Broadcast. 0 - Broadcast packets are forwarded to all ports. 1 - The GT-48001 discards Broadcast packets.	0x0

Bits	Field Name	Function	Initial Value
11	ForwUnk	Forward Unknown. 0 - Unknown packets are forwarded. 1 - The GT-48001 does not forward Unknown packets to this port.	0x0
12	SpanEn	Spanning Tree Enable. Meaningful only when SpanEn bit in the Global Control register is set. 0 - All packets are accepted. 1 - The GT-48001 discards all incoming/outgoing packets except for BPDU packets.	0x0

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

Bits	Field Name	Function	Initial Value
4:0	DevNum	Device Number. Indicates the GT-48001 number chosen by the designer.	DAddr[4:0] at reset
5	DRAMSize	DRAM Size. Indicates the DRAM size. 0- 2Mbyte 1- 1Mbyte	DAddr[5] 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 1 - Link Test Fail	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 1 - Link Test Fail	0x0
10	Port2Par	Port2 Partition. This bit indicates the port 2 Partition status. 0 - No Partition (Normal mode) 1 - Partition	0x0
11	Port2LTF	Port2 Link Test Fail. This bit indicates the port 2 Link Test status. 0 - Link Test Pass 1 - Link Test Fail	0x0

Bits	Field Name	Function	Initial Value
12	Port3Par	Port3 Partition. This bit indicates the port 3 Partition status. 0 - No Partition (Normal mode) 1 - Partition	0x0
13	Port3LTF	Port3 Link Test Fail. This bit indicates the port 33 Link Test status. 0 - Link Test Pass 1 - Link Test Fail	0x0
14	Port4Par	Port4 Partition. This bit indicates the port 4 Partition status. 0 - No Partition (Normal mode) 1 - Partition	0x0
15	Port4LTF	Port4 Link Test Fail. This bit indicates the port 4 Link Test status. 0 - Link Test Pass 1 - Link Test Fail	0x0
16	Port5Par	Port5 Partition. This bit indicates the port 5 Partition status. 0 - No Partition (Normal mode) 1 - Partition	0x0
17	Port5LTF	Port5 Link Test Fail. This bit indicates the port 5 Link Test status. 0 - Link Test Pass 1 - Link Test Fail	0x0
18	Port6Par	Port6 Partition. This bit indicates the port 6 Partition status. 0 - No Partition (Normal mode) 1 - Partition	0x0
19	Port6LTF	Port6 Link Test Fail. This bit indicates the port 6 Link Test status. 0 - Link Test Pass 1 - Link Test Fail	0x0
20	Port7Par	Port7 Partition. This bit indicates the port 7 Partition status. 0 - No Partition (Normal mode) 1 - Partition	0x0
21	Port7LTF	Port7 Link Test Fail. This bit indicates the port 7 Link Test status. 0 - Link Test Pass 1 - Link Test Fail	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 chosen by the designer.	0x0
7:5	SnifPortNum	Sniffer Port Number. These bits specify the Sniffer Port Number chosen by the designer.	0x0
12:8	MgmtDevNum	Management Device Number. These bits specify the Management Device Number chosen by the designer.	0x0

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-48001 when a new address is received.	0x0
2	TxEnd	Tx End of Packet. This bit is set by the GT-48001 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-48001 upon sending a 'start_of_packet' message to the CPU	0x0
4	AddrRecF	Address Recognition Failed. This bit is set by the GT-48001 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-48001 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-48001 upon master read parity error on the PCI.	0x0
7	MastWrPar	Master Write Parity. This bit is set by the GT-48001 upon master write error on the PCI.	0x0
8	AddPar	Address Parity. This bit is set by the GT-48001 upon address parity error on the PCI.	0x0
9	MastAbort	Master Abort. the This bit is set by the GT-48001 upon master abort on PCI.	0x0
10	TarAbort	Target Abort. This bit is set by the GT-48001 upon target abort on the PCI.	0x0

Bits	Field Name	Function	Initial Value
11	LTF	Link Test Fail. This bit is set by the GT-48001 upon Link Test Fail in one of the ports.	0x0
12	Part	Partition. This bit is set by the GT-48001 upon entering Partition state in one of the ports.	0x0
13	BufWrap	Buffer Wrap-Around. This bit is set by the GT-48001 upon transferring 16 packets to the CPU main memory.	0x0
14	Interv	Intervention. This bit is set by the GT-48001 upon transferring a 'buffer_request' message in Intervention mode.	0x0
15	IntervWrap	Intervention Wrap-Around. This bit is set by the GT-48001 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
17:0	-	Reserved.	0x0
24:18	IPGData	Inter-Packet Gap (IPG) Data. The step is 100ns (1 bit-time). The JAM IPG varies between 1 bit-time and 128 bit-times. The default value is 96 decimal, or 9.6us.	096d
31:25	DataBlind	Data Blinder. The range is 0 to 96. The default is 60 decimal.	060d

Rx Buffers Threshold, Offset: 0x040224

Bits	Field Name	Function	Initial Value
3:0	TxWatTim	Tx Watchdog Timer. The step is 10ms, the value varies between 10ms (0x1) and 160ms (0x10). The default is 60ms (0x6). (0x0 has no meaning, illegal condition.)	0x6
8:4	Reserve	Reserved.	0x00

Bits	Field Name	Function	Initial Value
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 step is 10, and the value varies between 10 (0x0) and 160 (0xf). The default is 40 (0x3) for 1MB DRAM and 80 (0x7) for 2MB DRAM.	0x3 (40) w/ 1M DRAM 0x7 (80) w/ 2M DRAM
16:13	Reserve	Reserved.	0x0

CPU Buffer Base Address, Offset: 0x140034

Bits	Field Name	Function	Initial Value
31:15	BaseAdd	Contains a pointer to the CPU buffer area	0x0
14:0	Reserve	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 (2 32-bit word each) for the GT-48001's 'start_of_packet' messages.	-
7:0	Reserve	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-48001's 'new_address' messages.	0x0
7:0	Reserve	Reserved.	-

CPU Intervention Base Address, Offset: 0x140048

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-48001's 'buffer_request' messages.	0x0
10:0	Reserve	Reserved.	-

Device Table, Offset: 0x40

Bits	Field Name	Function	Initial Value
31:0	DevTab	GT-48001 Device Table. Each bit represents a GT-48001 device in the system.	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-48001 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-48001 holds the PCI bus before the generation of Retry termination. Used for data transfers following the first data.	0x07 (8 clocks)

Port Counters (8 Blocks), Offset: 0x040000 - 0x0401fc

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	BytRec	Good Bytes Received.	-
0x040004	MulBytRec	Good Multicast Bytes Received.	-
0x040008	BroadBytRec	Good Broadcast Bytes Received.	-
0x04000c	BytSent	Good Bytes Sent.	-
0x040010	FraRec	Good Frames Received.	-
0x040014	MulFraRec	Good Multicast Frames Received.	-
0x040018	BroadFraRec	Broadcast Frames Received.	-
0x04001c	FraSent	Frames Sent.	-
0x040020	Coll	Receive and Transmit Collision.	-
0x040024	LateColl	Receive and Transmit Late Collision.	-
0x040028	CRCAligErr	Receive CRC or Alignment Error.	-
0x04002c	Jabber	Receive Frame > 1518 bytes with Bad CRC	-
0x040030	FraShort	Receive Fragments	-
0x040034	FraLong	Receive Frame > 1518 bytes with Good CRC	-
0x040038	BadBytRec	Bad Bytes Received. Bytes with CRC errors and frame too long.	-

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	--

3.3 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-48001 (0x11ab).	0x11ab
31:16	DevId	Provides the unique GT- 48001 ID number .	0x4801

Status and Command, Offset: 0x004

Bits	Field Name	Function	Initial Value
1	MemEn	Memory Enable. Controls the GT-48001's response to memory accesses, as found in the PCI specification. 0 - Disable 1 - Enable	0x1
2	MasEn	Master Enable. Controls the GT-48001'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-48001'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-48001'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-48001'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-48001 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-48001 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-48001's DevSel* timing. The GT-48001'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-48001 revision number.	0x0
23:16	SubClass	SubClass. Indicates the GT-48001 Subclass (0x0 - Ethernet).	0x0
31:24	BaseClass	Base Class. Indicates the GT-48001 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-48001'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-48001. 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.	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.	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-48001 uses. The GT-48001 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	AD[4]	36	DData[5]	71	CrS/RxEn[6]
2	AD[3]	37	DData[4]	72	Pol[6]
3	AD[2]	38	DData[3]	73	TxEn[6]
4	AD[1]	39	DData[2]	74	TxD[6]
5	AD[0]	40	DData[1]	75	TxDDel[6]
6	DData[31]	41	DData[0]	76	RxD[5]
7	DData[30]	42	VDD	77	RxLP/Coll[5]
8	DData[29]	43	VSS	78	CrS/RxEn[5]
9	DData[28]	44	CAS*	79	Pol[5]
10	DData[27]	45	WE*	80	TxEn[5]
11	DData[26]	46	RAS[1]*	81	TxD[5]
12	DData[25]	47	RAS[0]*	82	TxDDel[5]
13	DData[24]	48	ChipSel*	83	RxD[4]
14	DData[23]	49	DAddr[8]	84	RxLP/Coll[4]
15	DData[22]	50	DAddr[7]	85	CrS/RxEn[4]
16	DData[21]	51	DAddr[6]	86	Pol[4]
17	DData[20]	52	DAddr[5]	87	TxEn[4]
18	DData[19]	53	DAddr[4]	88	TxD[4]
19	DData[18]	54	DAddr[3]	89	TxDDel[4]
20	VDD	55	DAddr[2]	90	VDD
21	VSS	56	DAddr[1]	91	SClk/SynClk20
22	DData[17]	57	DAddr[0]	92	VSS
23	DData[16]	58	VDD	93	VDD
24	VDD	59	VSS	94	VSS
25	VSS	60	RxD[7]	95	RxD[3]
26	DData[15]	61	RxLP/Coll[7]	96	RxLP/Coll[3]
27	DData[14]	62	CrS/RxEn[7]	97	CrS/RxEn[3]
28	DData[13]	63	Pol[7]	98	Pol[3]
29	DData[12]	64	TxEn[7]	99	TxEn[3]
30	DData[11]	65	TxD[7]	100	TxD[3]
31	DData[10]	66	TxDDel[7]	101	TxDDel[3]
32	DData[9]	67	VDD	102	RxD[2]
33	DData[8]	68	VSS	103	RxLP/Coll[2]
34	DData[7]	69	RxD[6]	104	CrS/RxEn[2]
35	DData[6]	70	RxLP/Coll[6]	105	Pol[2]

Pin #	Signal Name	Pin #	Signal Name	Pin #	Signal Name
106	TxEn[2]	141	VSS	176	AD[17]
107	TxD[2]	142	VDD	177	AD[16]
108	TxDDel[2]	143	VSS	178	CBE[2]*
109	RxD[1]	144	Int*	179	Frame*
110	RxLP/Coll[1]	145	Rst*	180	VSS
111	CrS/RxEn[1]	146	VDD	181	VDD
112	Pol[1]	147	Clk	182	IRdy*
113	TxEn[1]	148	VSS	183	TRdy*
114	TxD[1]	149	Gnt*	184	DevSel*
115	TxDDel[1]	150	Req*	185	Stop*
116	RxD[0]	151	VSS	186	PErr*
117	RxLP/Coll[0]	152	VDD	187	VSS
118	CrS/RxEn[0]	153	AD[31]	188	VDD
119	Pol[0]/SynClk10	154	AD[30]	189	SErr*
120	TxEn[0]	155	AD[29]	190	Par
121	TxD[0]	156	AD[28]	191	CBE[1]*
122	TxDDel[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	AD[13]
126	RstQueue*	161	AD[25]	196	AD[12]
127	VSS	162	AD[24]	197	AD[11]
128	VSS	163	CBE[3]*	198	AD[10]
129	VSS	164	IdSel	199	AD[9]
130	DisWD*	165	VSS	200	VSS
131	EnELScrub*	166	VDD	201	VDD
132	SkipInit*	167	AD[23]	202	AD[8]
133	Scan*	168	AD[22]	203	CBE[0]*
134	TriState*	169	AD[21]	204	AD[7]
135	VSS	170	AD[20]	205	AD[6]
136	LEDClk	171	AD[19]	206	AD[5]
137	LEDData	172	VSS	207	VSS
138	LEDStb	173	VDD	208	VDD
139	VSS	174	VSS		
140	VDD	175	AD[18]		

5 DC CHARACTERISTICS - *PRELIMINARY/SUBJECT TO CHANGE*

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
Tc	Operating Case Temperature	0	70	C
Tstg	Storage Temperature	-40	125	C
ESD			2000	V

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
Tc	Operating Case Temperature	0		70	C
Cin	Input Capacitance		7.2		pF
Cout	Output Capacitance		7.2		pF

5.3 DC Electrical Characteristics Over Operating Range

(Tc=0-70C; 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	Vdd=5.25V, f=33MHz			600	mA

NOTE:

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

5.4 Package Thermal Characteristics

Symbol	Max.	Unit
Θjc	12	C/W
Θja	40	C/W

Galileo Technology recommends all GT-48001 systems to use a heat sink to maintain Tc of 70C. Additional airflow may be necessary in some enclosures.

6 AC TIMING - *PRELIMINARY/SUBJECT TO CHANGE*

(TC= 0-70°C; VDD= +5V, +/- 5%)

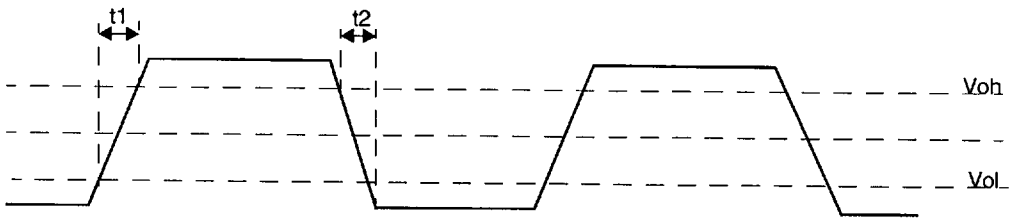
Symbol	Signals	Description	Min	Max	Unit
	Clk	System/PCI Clock Frequency	30.0	33.0	MHz
	Rst*, Frame*, IRdy*, TRdy*, DevSel*, Stop*, PErr*, Par, Int* AD[31:0], CBE[3:0]*, Gnt*, IdSel, Req*, SErr*	See PCI Specification Rev. 2.1.			
t1	SClk	Rise Time		2	nS
t2	SClk	Fall Time		2	nS
	SClk	Frequency Stability		+/- 50	PPM
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:

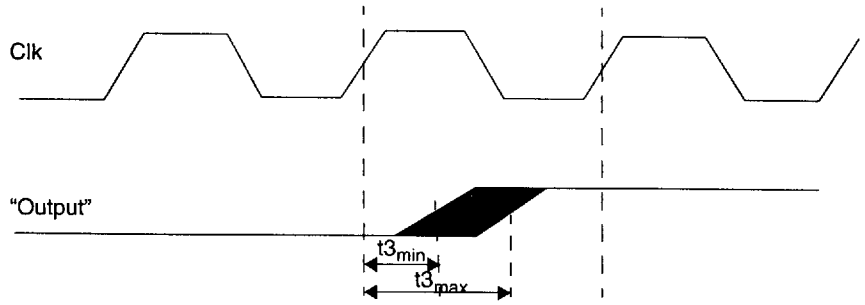
1. All Delays, Setup, and Hold times are referred to Clk rising edge, unless stated otherwise.

2. All outputs are specified for 50pf load.

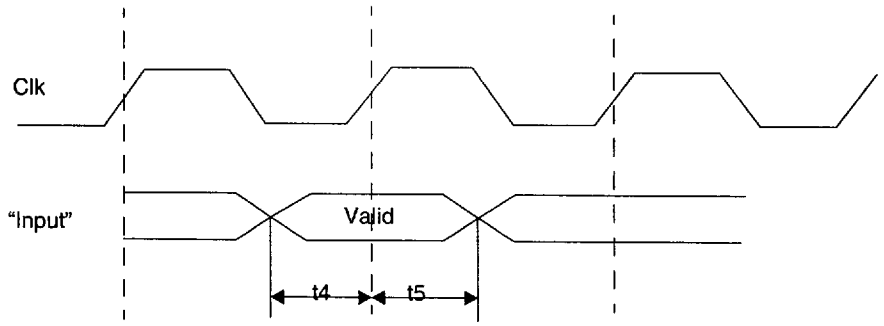
3. "All Inputs" and "All Outputs" also refer to I/O signals' behavior.



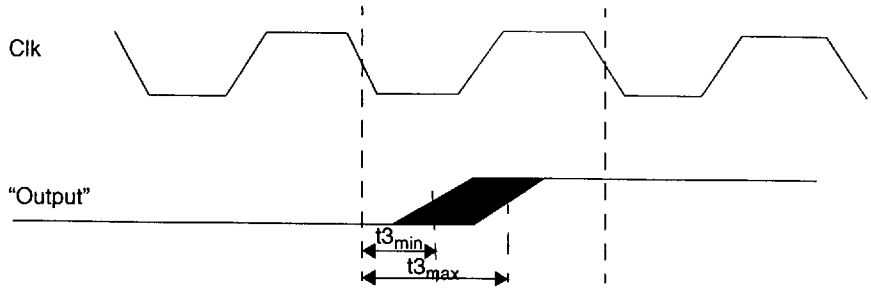
SClk



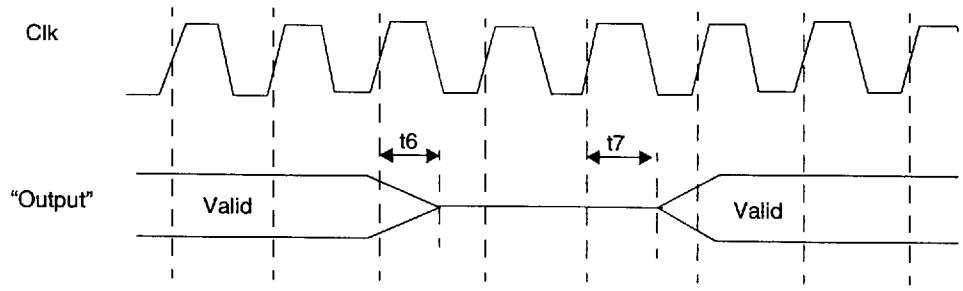
Output Delay from Clock Rising Edge



Inputs Setup and Hold



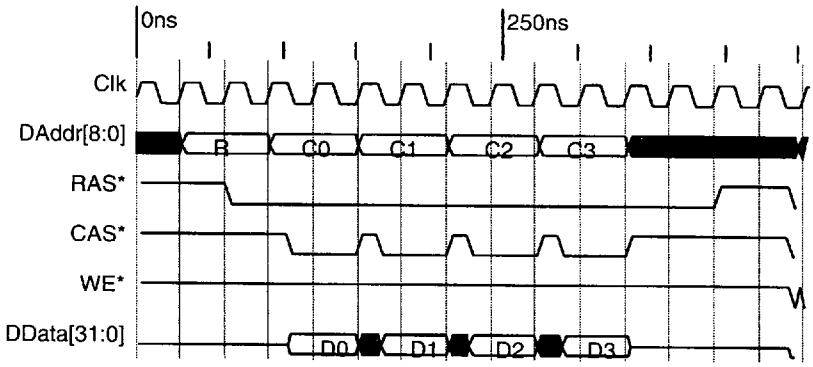
Output Delay from Clock Falling Edge



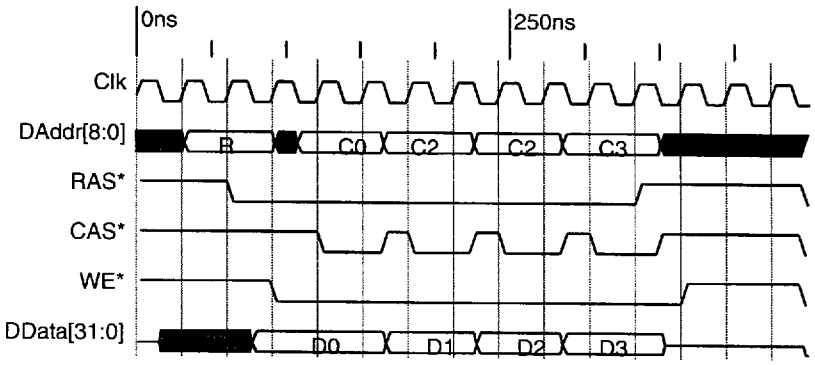
Output Float and Drive Delay

7 FUNCTIONAL WAVEFORMS

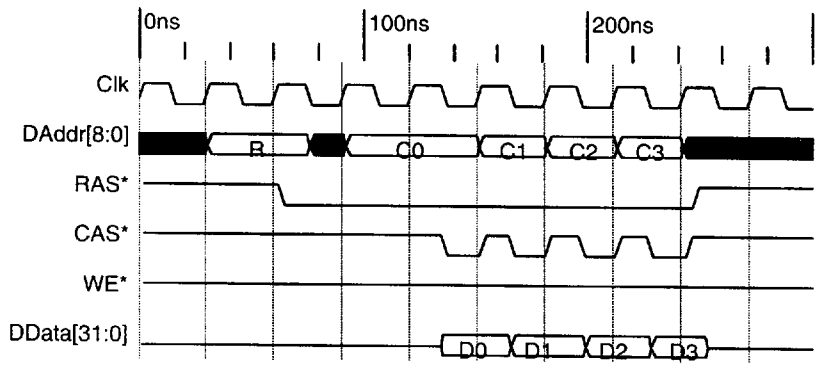
7.1 Standard DRAM Read



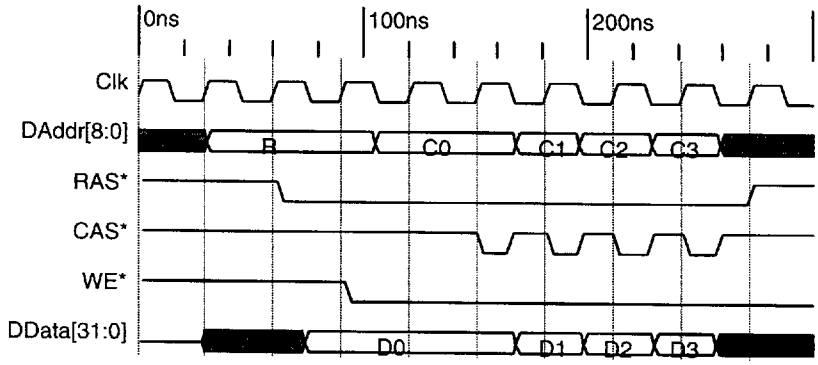
7.2 Standard DRAM Write



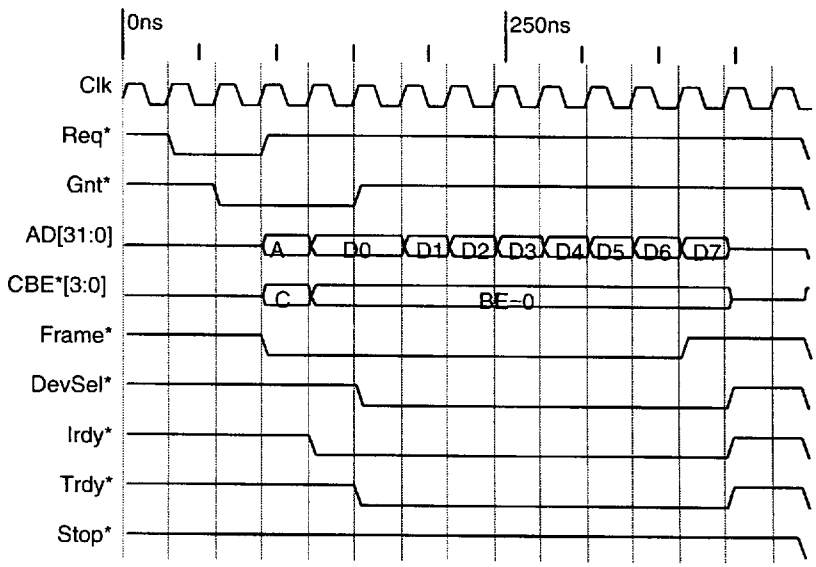
7.3 EDO Read



7.4 EDO Write



7.5 PCI Read/Write Cycle



8 SYSTEM CONFIGURATIONS

8.1 Switch Expansion

Figure 6 describes the way to expand the switch. Multiple GT-48001 devices are connected to a local PCI bus (up to 6). 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-48001 devices in one PCI bus is not forwarded to the other GT-48001 devices. Only the traffic between GT-48001 devices in different PCI buses crosses the bridges.

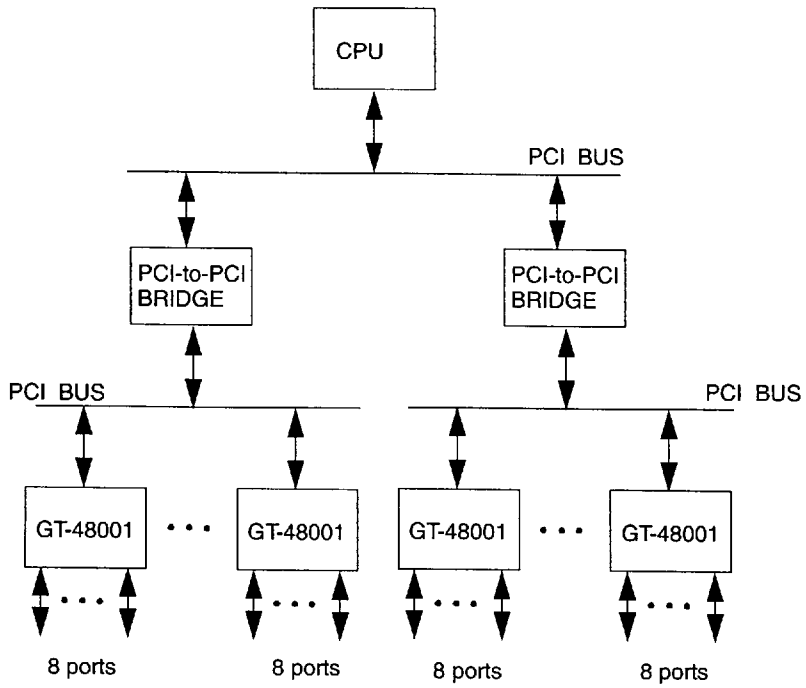
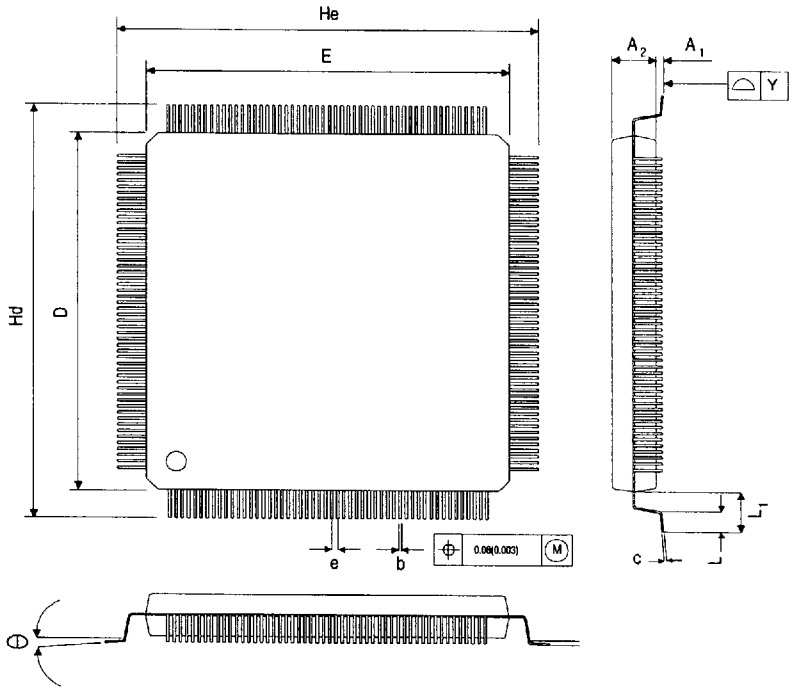


Figure 6 - Switch Expansion

9 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

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

10 Revision History

Document Type	Revision Number	Date	Comments
PRELIMINARY REV.	1.0	3/96	First revision of PRELIMINARY REVISION for general distribution.
PRELIMINARY REV.	1.1	12/96	DC Characteristics Revised Added DisWD*, EnELScrub*, SkipInit* Descriptions Fixed Errata listed in Documentation Update, Rev. 30