



STANDARD  
MICROSYSTEMS  
CORPORATION

## LAN83C171 ADVANCE INFORMATION

### LAN83C171 - EPIC/XF ACPI / PC 97 Compliant Integrated PCI 10/100Mb/s Ethernet Controller

#### FEATURES

- IEEE 802.3 Compatible 10/100 Mb/s Ethernet Controller
- Fully Compliant Glueless PCI Version 2.1 Bus Interface
- Support Included for CardBus Status Registers
- PCI Universal 3.3V/5V Output Drives
- Preemptive Interrupt Support for Efficient Network Packet Processing
- High Performance Two Channel Bus Master (132 Mbps)
- Scatter/Gather DMA Capability
- Programmable Burst Length Counter
- ACPI Compliant for
  - PCI Bus Class Specification
  - Network Device Class Specification
- PC 97 Compliant
- Wake-Up on Magic Packet™ Detection and/or Network Link-Down Occurrence
- Special Low Power State Mode For Scanning Magic Packets™ Upon PCI Bus Power Loss
- Supports Chaining of Transmit Packets
- Optional Early Transmit and Early Receive
- Optional Receive Lookahead Buffering Mode
- Automatic Rejection of Runt Packets
- Automatic Retransmission of Collision Frames from Internal Buffer
- Automatic Padding of Short Frames
- 4.5 Kbyte On-Chip Receive Buffer and 1.5 Kbyte On-Chip Transmit Buffer Eliminate Bus Latency Issues
- Optional Variable Depth, 32 Bit Wide External Receive Buffer (0, 16, 32 or 128 Kbytes)
- Big or Little Endian Byte Ordering
- Capable of Supporting 64 Kbyte Expansion Boot Flash RAM
- IEEE Standard MII Interface to Physical Layer
- Interface to LAN83C694 - Shares MII Pins
- Serial MII Management Interface
- Serial EEPROM Interface for Storage of LAN Address and Configuration Information
- On-Chip Clock Multiplier
- Low Power Sleep Mode
- Support for Full Duplex Ethernet
- Internal and External Loopback Diagnostic Functions
- Simple I/O Pin Mapping Scheme to Facilitate In-Circuit Test
- Single 5V Power Supply
- 208 Pin PQFP Package

## GENERAL DESCRIPTION

The LAN83C171 EPIC/XF is a high-performance and a low CPU utilization Ethernet network controller designed to interface directly to the PCI Local Bus on one side and to the 802.3 standard Media Independent Interface (MII) on the other side. The network interface can also be configured to communicate directly with the LAN83C694 10BASE-T transceiver.

The LAN83C171 implements 802.3 Media Access Control functions. It is capable of running at Ethernet rates of both 100Mb/s and

10Mb/s. An MII compliant serial management interface is provided to control external media dependent transceivers. The LAN83C171 is a two channel bus master (one for transmit, one for receive) capable of transferring data at the maximum PCI transfer rate of 132Mbps. The LAN83C171 has several features designed to minimize CPU utilization, including the optional Receive Lookahead Buffering Mode, which eliminates the need to re-copy the data from one host memory location to another.

## TABLE OF CONTENTS

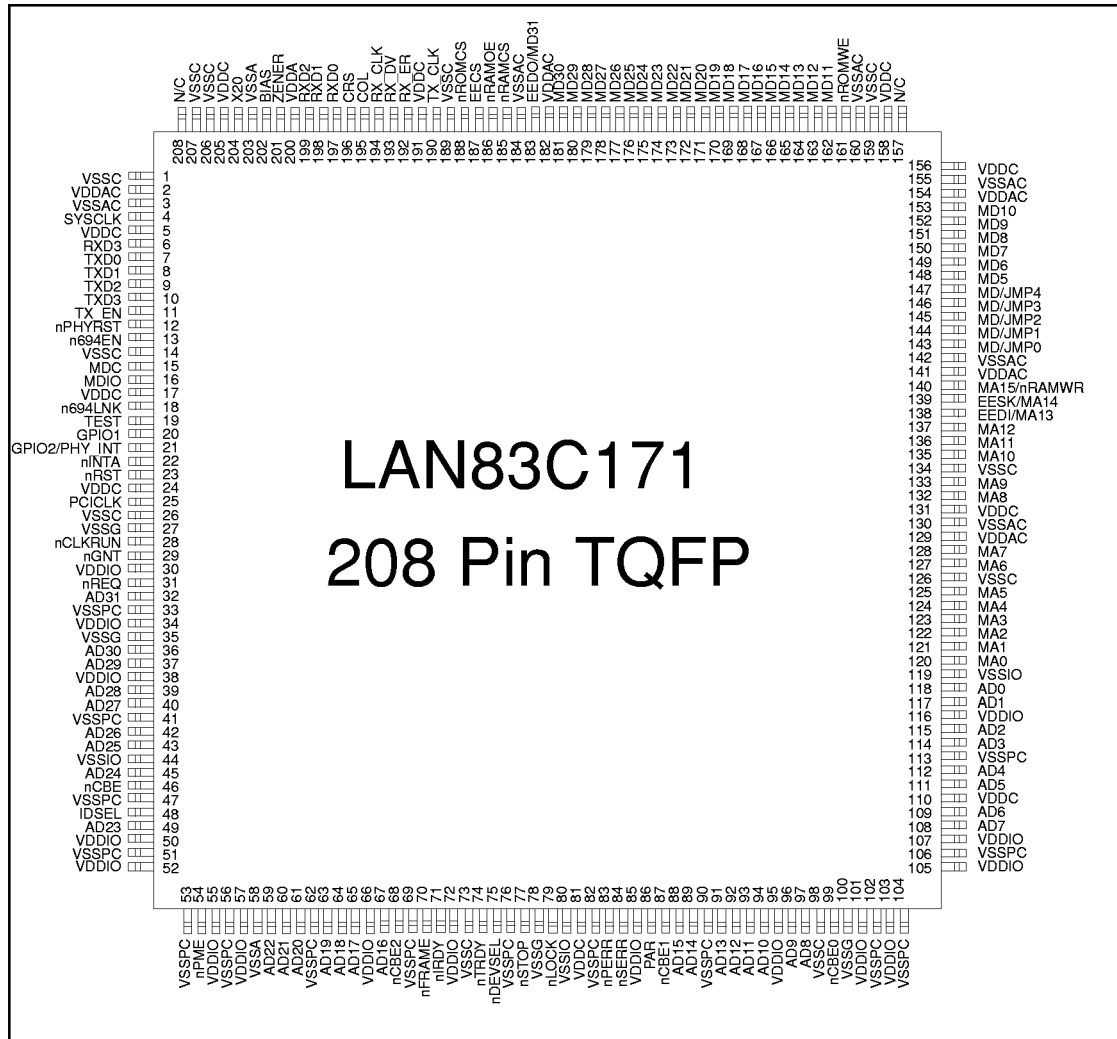
|                                                                |           |
|----------------------------------------------------------------|-----------|
| <b>FEATURES .....</b>                                          | <b>1</b>  |
| <b>GENERAL DESCRIPTION .....</b>                               | <b>2</b>  |
| <b>PIN CONFIGURATION.....</b>                                  | <b>5</b>  |
| <b>DESCRIPTION OF PIN FUNCTIONS .....</b>                      | <b>6</b>  |
| <b>FUNCTIONAL DESCRIPTION.....</b>                             | <b>10</b> |
| PCI INTERFACE.....                                             | 12        |
| TRANSMIT/RECEIVE ARBITRATION FOR PCI BUS .....                 | 12        |
| SYSTEM ERRORS .....                                            | 12        |
| BIG/LITTLE ENDIAN SUPPORT .....                                | 12        |
| <b>POWER DOWN MODE .....</b>                                   | <b>14</b> |
| DMA OPERATION .....                                            | 14        |
| <b>TRANSMIT DMA.....</b>                                       | <b>14</b> |
| Direct Queuing Method.....                                     | 14        |
| Fragment List Method .....                                     | 16        |
| Interrupting Transmit Chain.....                               | 18        |
| Transmit Buffer Full .....                                     | 18        |
| Transmit Underrun .....                                        | 18        |
| Exception to Underrun ReTransmission.....                      | 18        |
| Maximum Transmit Size and Burst Rate.....                      | 18        |
| <b>RECEIVE DMA.....</b>                                        | <b>19</b> |
| Free Buffer Pool Method.....                                   | 19        |
| Adding Receive Buffers to the Pool.....                        | 20        |
| Receive Lookahead Method .....                                 | 20        |
| Stopping the Receive DMA.....                                  | 25        |
| Maximum Receive Size and Burst Rate .....                      | 25        |
| <b>MAC OPERATION .....</b>                                     | <b>26</b> |
| MII MANAGEMENT INTERFACE .....                                 | 32        |
| EEPROM INTERFACE.....                                          | 32        |
| JUMPER OPTIONS (EEPROM/RAM).....                               | 33        |
| Advanced Configuration and Power Interface (ACPI) Support..... | 33        |
| Wake-Up Events and Notification .....                          | 33        |
| <b>EPIC Power States</b> .....                                 | 34        |
| D3(Cold1) Software Driver Requirements .....                   | 35        |
| Supporting Power Management Options.....                       | 35        |
| PME Generates a PCI Bus Interrupt .....                        | 35        |
| Initial Power-On Reset (POR).....                              | 35        |
| Special Power Management Mode.....                             | 36        |
| POWER DOWN MODE .....                                          | 36        |
| SOFT RESET .....                                               | 37        |
| <b>CONFIGURATION .....</b>                                     | <b>37</b> |
| Mapping of Control Functions.....                              | 37        |
| Mapping of Flash RAM Functions.....                            | 37        |
| <b>DMA DESCRIPTOR BITS DESCRIPTION.....</b>                    | <b>39</b> |
| TRANSMIT DMA DESCRIPTOR BITS DESCRIPTION .....                 | 39        |
| RECEIVE DMA DESCRIPTOR BITS DESCRIPTION .....                  | 41        |
| <b>CONTROL REGISTER MAP/REGISTERS DECODE.....</b>              | <b>43</b> |
| <b>CONFIGURATION REGISTERS MAP.....</b>                        | <b>44</b> |

|                                                          |           |
|----------------------------------------------------------|-----------|
| <b>CONTROL REGISTERS BITS DESCRIPTION .....</b>          | <b>45</b> |
| <b>PCI CONFIGURATION REGISTERS BITS DESCRIPTION.....</b> | <b>64</b> |
| <b>OPERATIONAL DESCRIPTION.....</b>                      | <b>70</b> |
| Maximum Guaranteed Ratings.....                          | 70        |
| DC Electrical Characteristics .....                      | 70        |
| <b>TIMING DIAGRAMS .....</b>                             | <b>72</b> |



80 Arkay Drive  
 Hauppauge, NY 11788  
 (516) 435-6000  
 FAX (516) 273-3123

## PIN CONFIGURATION



### DESCRIPTION OF PIN FUNCTIONS

| PQFP PIN NO.                                                                                                    | NAME    | I/O | TYPE                          | DESCRIPTION                                                                                         |
|-----------------------------------------------------------------------------------------------------------------|---------|-----|-------------------------------|-----------------------------------------------------------------------------------------------------|
| <b>PCI INTERFACE</b>                                                                                            |         |     |                               |                                                                                                     |
| 25                                                                                                              | PCICLK  | I   | I <sub>PClk</sub> ,<br>dc_1k1 | PCI Clock                                                                                           |
| 23                                                                                                              | nRST    | I   | I <sub>PCI</sub> ,<br>dc_1k1  | PCI System Reset                                                                                    |
| 32,36,37,39,40,42,<br>43,45,49,59-61,<br>63-65,67,88,89,<br>91-94,96,97,108,<br>109,111,112,114,<br>115,117,118 | AD      | I/O | IO <sub>PCI</sub> ,<br>dc_1k2 | PCI Multiplexed Address/Data Bus                                                                    |
| 46,68,87,99                                                                                                     | nCBE    | I/O | IO <sub>PCI</sub> ,<br>dc_1k2 | PCI Multiplexed Command/Byte Enable Signals                                                         |
| 86                                                                                                              | PAR     | I/O | IO <sub>PCI</sub> ,<br>dc_1k2 | PCI Parity Signal                                                                                   |
| 70                                                                                                              | nFRAME  | I/O | IO <sub>PCI</sub> ,<br>dc_1k2 | PCI Cycle Frame Signal                                                                              |
| 71                                                                                                              | nIRDY   | I/O | IO <sub>PCI</sub> ,<br>dc_1k2 | PCI Initiator Ready Signal                                                                          |
| 74                                                                                                              | nTRDY   | I/O | IO <sub>PCI</sub> ,<br>dc_1k2 | PCI Target Ready Signal                                                                             |
| 77                                                                                                              | nSTOP   | I/O | IO <sub>PCI</sub> ,<br>dc_1k2 | PCI Cycle Stop Signal                                                                               |
| 79                                                                                                              | nLOCK   | I   | IO <sub>PCI</sub> ,<br>dc_1k2 | PCI Lock Signal                                                                                     |
| 48                                                                                                              | IDSEL   | I   | I <sub>PCI</sub> ,<br>dc_1k1  | PCI Initiation Device Select Signal.<br>Used as a Chip Select for<br>Configuration Reads and Writes |
| 75                                                                                                              | nDEVSEL | I/O | IO <sub>PCI</sub> ,<br>dc_1k2 | PCI Device Select                                                                                   |
| 31                                                                                                              | nREQ    | O   | O <sub>PCI</sub> ,<br>dc_1k2  | PCI Bus Request                                                                                     |
| 29                                                                                                              | nGNT    | I   | I <sub>PCI</sub> ,<br>dc_1k1  | PCI Bus Grant                                                                                       |
| 83                                                                                                              | nPERR   | I/O | IO <sub>PCI</sub> ,<br>dc_1k2 | PCI Parity Error                                                                                    |
| 84                                                                                                              | nSERR   | O   | O <sub>PCI</sub> ,<br>dc_1k2  | PCI System Error (Open Drain)                                                                       |

| PQFP PIN NO.                        | NAME        | I/O | TYPE                           | DESCRIPTION                                                                                            |
|-------------------------------------|-------------|-----|--------------------------------|--------------------------------------------------------------------------------------------------------|
| 22                                  | nINTA       | O   | Open Drain<br>dc_1k2           | PCI Interrupt (Open Drain)                                                                             |
| 54                                  | nPME        | O   | Open Drain<br>dc_1k2           | PCI Power Management Event (Open Drain)                                                                |
| 28                                  | nCLKRUN     | I/O | IO <sub>PCI</sub> ,<br>dc_1k2  | PCI Clock Control and Request Line                                                                     |
| <b>EXTERNAL MEMORY INTERFACE</b>    |             |     |                                |                                                                                                        |
| 140                                 | MA15/nRAMWR | O   | O <sub>TTL8</sub> ,<br>dc_1k2  | Most Significant Address Bit to Expansion ROM/External RAM Write Enable                                |
| 139-135,133,132,<br>128,127,125-120 | MA          | O   | O <sub>TTL8</sub> ,<br>dc_1k2  | Address Bus to Shared RAM and ROM. (EEDI Muxed Into MA[13], EESK Muxed Into MA[14])                    |
| 183,181-162,<br>153-143             | MD/JMP      | I/O | IO <sub>TTL2</sub> ,<br>dc_1k2 | Data Bus to Shared RAM, ROM and EEPROM (EEDO Input Connected to MD[31], JMP[4:0] Connected to MD[4:0]) |
| 186                                 | nRAMOE      | O   | O <sub>TTL8</sub> ,<br>dc_1k2  | External RAM Output Enable                                                                             |
| 185                                 | nRAMCS      | O   | O <sub>TTL8</sub> ,<br>dc_1k2  | External RAM Chip Select                                                                               |
| 188                                 | nROMCS      | O   | O <sub>TTL4</sub> ,<br>dc_1k2  | ROM Chip Select and Output Enable                                                                      |
| 161                                 | nROMWE      | O   | O <sub>TTL4</sub> ,<br>dc_1k2  | External Flash ROM Write Select                                                                        |
| 187                                 | EECS        | O   | O <sub>TTL4</sub> ,<br>dc_1k2  | EEPROM Chip Select                                                                                     |
| <b>NETWORK INTERFACE</b>            |             |     |                                |                                                                                                        |
| 190                                 | TX_CLK      | I   | IO <sub>TTL4</sub> ,<br>dc_1k2 | MII Transmit Clock                                                                                     |
| 11                                  | TX_EN       | O   | O <sub>TTL4</sub> ,<br>dc_1k2  | MII Transmit Enable                                                                                    |
| 10-7                                | TXD         | O   | O <sub>TTL4</sub> ,<br>dc_1k2  | MII Transmit Data                                                                                      |
| 194                                 | RX_CLK      | I   | IO <sub>TTL4</sub> ,<br>dc_1k2 | MII Receive Clock                                                                                      |

| PQFP PIN NO.                           | NAME          | I/O | TYPE                           | DESCRIPTION                                                                                                                                |
|----------------------------------------|---------------|-----|--------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| 196                                    | CRS           | I   | IO <sub>TTL4</sub> ,<br>dc_1k2 | MII Carrier Sense                                                                                                                          |
| 6,199-197                              | RXD           | I   | IO <sub>TTL4</sub> ,<br>dc_1k2 | MII Receive Data                                                                                                                           |
| 195                                    | COL           | I   | IO <sub>TTL4</sub> ,<br>dc_1k2 | MII Collision Signal                                                                                                                       |
| 193                                    | RX_DV         | I   | IO <sub>TTL4</sub> ,<br>dc_1k2 | MII Receive Data Valid Signal                                                                                                              |
| 192                                    | RX_ER         | I   | IO <sub>TTL4</sub> ,<br>dc_1k2 | MII Receive Error Signal                                                                                                                   |
| 15                                     | MDC           | O   | O <sub>TTL4</sub> ,<br>dc_1k2  | MII Management Interface Clock                                                                                                             |
| 16                                     | MDIO          | I/O | IO <sub>TTL4</sub> ,<br>dc_1k2 | MII Management Interface Data                                                                                                              |
| 18                                     | n694LNK       | I   |                                | 10Base-T Link Integrity Status                                                                                                             |
| 13                                     | n694EN        | O   |                                | 694 Enable. Tri-States 694 Outputs when PHY100 is in use                                                                                   |
| <b>GENERAL PURPOSE</b>                 |               |     |                                |                                                                                                                                            |
| 4                                      | SYSCLK        | I   | I <sub>SCLK</sub>              | System Clock (25 MHz) Input                                                                                                                |
| 20                                     | GPIO1         | I/O |                                | General Purpose I/O. May also be used to indicate if the source of power is the PCI Bus. High(1) to indicate the power source from PCI bus |
| 21                                     | GPIO2/PHY_INT | I/O |                                | General Purpose I/O. Can also be used as a physical layer interrupt. It should be passively pulled high                                    |
| 19                                     | TEST          | I   | I <sub>TTL</sub>               | Used for In-Circuit Device Test                                                                                                            |
| 204                                    | X20           | O   | O <sub>TTL4</sub> ,<br>dc_1k2  | 20 MHz Buffered Clock Output                                                                                                               |
| 202                                    | BIAS          | I   | I <sub>AN</sub>                | Bias Current Input for Clock Multiplier                                                                                                    |
| 201                                    | ZENER         | I   | I <sub>AN</sub>                | Regulated Voltage Input for Clock Multiplier                                                                                               |
| 12                                     | nPHYRST       | O   |                                | Reset Output to Physical Layer Chip                                                                                                        |
| <b>POWER SUPPLY</b>                    |               |     |                                |                                                                                                                                            |
| 5,17,24,81,110,131,<br>156,158,191,205 | VDDC          | PWR |                                | Connect to +5V                                                                                                                             |

| PQFP PIN NO.                                                     | NAME  | I/O | TYPE | DESCRIPTION       |
|------------------------------------------------------------------|-------|-----|------|-------------------|
| 2,129,141,154,182                                                | VDDAC | PWR |      | Connect to +5V    |
| 34,50,72,101,30,<br>38,52,55,85,95,<br>103,105,116,57,66,<br>107 | VDDIO | PWR |      | Connect to +5V    |
| 200                                                              | VDDA  | PWR |      | Connect to +5V    |
| 1,14,26,73,98,126,<br>134,159,189,206,<br>207                    | VSSC  | PWR |      | Connect to Ground |
| 3,130,142,155,160,<br>184                                        | VSSAC | PWR |      | Connect to Ground |
| 27,35,78,100                                                     | VSSG  | PWR |      | Connect to Ground |
| 33,41,47,51,53,56,<br>62,69,76,82,90,<br>102,104,106,113         | VSSPC | PWR |      | Connect to Ground |
| 44,80,119                                                        | VSSIO | PWR |      | Connect to Ground |
| 58,203                                                           | VSSA  | PWR |      | Connect to Ground |

## FUNCTIONAL DESCRIPTION

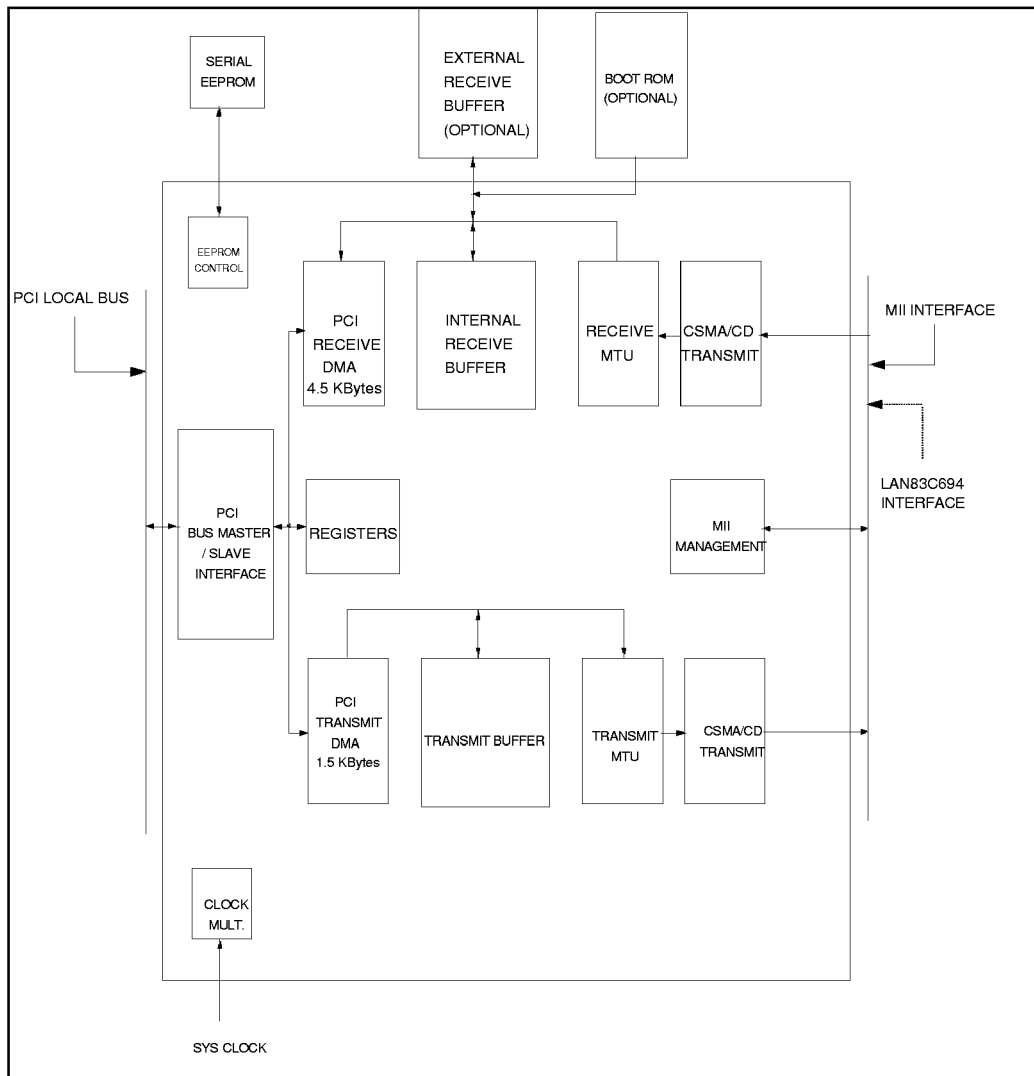
The LAN83C171 EPIC/100 is a high performance Ethernet network controller designed to interface directly to the PCI Local Bus on one side and to the 802.3 standard Media Independent Interface (MII) on the other side. The network interface can also be configured to communicate directly with the LAN83C694 10BASE-T transceiver.

The LAN83C171 implements 802.3 Media Access Control functions. It is capable of running at Ethernet rates of both 100 Mb/s and 10 Mb/s. An MII compliant serial management interface is provided to control external media dependent transceivers. The LAN83C171 is a two channel bus master (one for transmit, one for receive) capable of transferring data at the maximum PCI transfer rate of 132 Mbps. Buffer format in host memory is controlled by an independent linked list structure for each channel.

The LAN83C171's architecture is essentially broken into two independent transmit and receive processes which share PCI bus and network bandwidth. This architecture is ideal for full-duplex networks where transmission and reception of frames may occur simultaneously. An internal arbiter controls which process has access to the PCI bus at a given time (see section on "transmit/receive arbitration for PCI bus").

The transmit process consists of a DMA controller, local transmit RAM, memory transfer unit ("MTU") and CSMA/CD transmit state machine. The transmit DMA copies packet data from host memory into the local buffer. When ready, the memory transfer unit feeds data from the transmit buffer to the CSMA/CD state machine, which is responsible for sending data out on the network under the Ethernet protocol. When transmission is complete, the transmit DMA posts the transmit status into host memory, interrupts the host (optionally) and looks for the next transmit packet to be queued.

Like the transmit process, the receive process consists of a DMA controller, local receive RAM, memory transfer unit and CSMA/CD state machine. Packets are received by the CSMA/CD state machine and stored into local memory by the receive MTU. The receive DMA then copies the data from the local buffer into host memory, posts the receive status and interrupts the host. The LAN83C171 has several features designed to minimize CPU utilization, including the optional Receive Lookahead Buffering Mode, which eliminates the need to re-copy the data from one host memory location to another. Figure 1 on the following page shows a block diagram of the LAN83C171.



**FIGURE 1 - LAN83C171 BLOCK DIAGRAM**

## PCI INTERFACE

The transmit and receive DMA controllers access the PCI bus through the PCI bus Master/Slave Interface logic. This block is responsible for requesting the PCI bus and conducting all bus master operations according to the PCI bus protocol (including parity generation and error detection). This block is also responsible for responding to all slave operations according to PCI bus protocol (including address recognition, parity generation, and error detection).

### TRANSMIT/RECEIVE ARBITRATION FOR PCI BUS

Another major function of the PCI Bus Master/Slave Interface block is to arbitrate between the transmit and receive DMA controllers for access to the PCI bus.

### SYSTEM ERRORS

There are four types of PCI bus errors that are considered fatal by the LAN83C171. They are Master Abort, Target Abort, Address Parity Error, and Data Parity Error (see interrupt status register for details). If any of these errors occurs, the LAN83C171 will set the appropriate interrupt and immediately discontinue all DMA activity. The receiver will automatically be taken offline and any transmissions in progress will be completed without a valid CRC appended (in case transmit data was corrupted). Normal operation may only be resumed by resetting the LAN83C171 with the soft reset bit. The software driver should make sure the transmitter and receiver have returned to their idle states (by polling the TXIDLE and RXIDLE bits in the interrupt status register) before resetting the device.

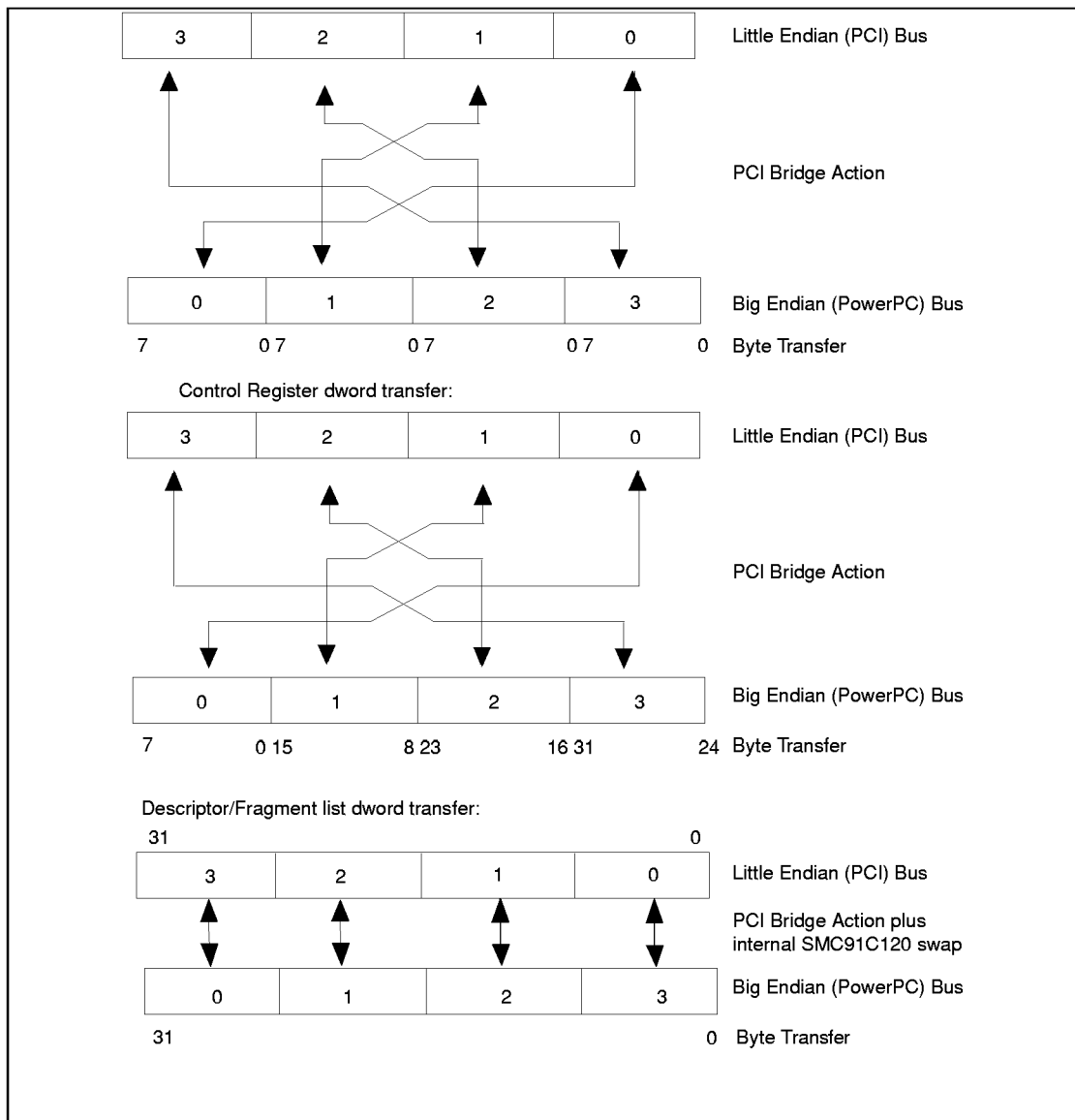
## BIG/LITTLE ENDIAN SUPPORT

In order to run in Big Endian machines, the LAN83C171 can be programmed to swap bytes on the data bus in certain circumstances. In Macintosh Power PC computers the bridge between the Big Endian processor data bus and the Little Endian PCI bus swaps the order of the bytes on the data bus (during data phase only - addresses are never modified). This means that byte size quantities transferred over the data bus always end up in the correct location for their given address, but when 32 bit (dword) quantities are transferred they end up with their bytes reversed.

When programmed into Big Endian mode, the LAN83C171 automatically swaps the data bytes internally when reading or writing descriptor tables or fragment lists. This allows the software driver to treat the descriptor and fragment list entries as 32 bit quantities and not worry about byte ordering.

In order to comply with the PCI specification, the LAN83C171 does not swap the data bytes on reads or writes to the configuration or control register space. The software driver is responsible for interpreting correctly the bytes when performing 32 bit register read or writes on a Big Endian machine.

When reading or writing Ethernet packet data, the LAN83C171 does not perform any byte swapping internally because the data on the PCI bus is already in the correct order.



**FIGURE 2 - LITTLE ENDIAN/BIG ENDIAN BYTE TRANSFER**

The number in the box refers to the address of the byte. The numbers above and below the boxes refer to the bit number (the bit

ordering increases from LSB to MSB for both formats although some other documents choose to label them differently).

## POWER DOWN MODE

The LAN83C171 has a power down feature which allows it to consume less power when not in use. The host may power down the LAN83C171 by writing a 1 to the power down bit in the general control register. When the bit is set, the chip's internal system clock is gated off to reduce switching current (the transmit and receive clocks will be shut off internally if the LAN83C171 is in loopback mode when power down is set). While the LAN83C171 is powered down, the host may read and write the configuration registers or the general control register. All other functions are disabled (attempting any other operation will cause unpredictable behavior). The power down bit must only be set when the LAN83C171 is in its idle state.

When the nRST pin is asserted, the LAN83C171 will automatically enter power down mode after recalling the contents of the EEPROM. The host may power up the LAN83C171 by writing a 0 to the power down bit. If the host wishes to issue a software reset to the LAN83C171, the power down bit must be cleared. When the software reset has completed, the power down bit will remain cleared and the LAN83C171 will be ready to operate.

The power down bit does not affect the PCI clock inside the LAN83C171. Instead, the LAN83C171 supports the PCI clock run function which allows the host system to slow down or temporarily shut off the PCI clock at its source. The clock run function is implemented according to the PCI Mobile design guide (revision 1.0).

## DMA OPERATION

The software driver controls the transmit and receive DMA controllers through the I/O control registers and through "buffer descriptors" in host memory. There is an

independent chain (linked list) of descriptors for each DMA. Each descriptor may point to a single data buffer (which can hold a whole frame or part of a frame) or to a fragment list, which in turn contains a list of buffers for an entire frame. Each descriptor also contains control and status information and a pointer to the next descriptor. The Descriptors Bit Description section explains the bits in detail.

## TRANSMIT DMA

The software driver initializes the transmit process by writing the transmit control register, early transmit threshold register (if early transmit will be used), interpacket gap program register, interrupt mask register and general control register. The software must also program the PCI Transmit Current Descriptor Address Register (PTCDAR) with the address in host memory where the first transmit descriptor will be located.

To begin packet transmissions, the software driver programs the transmit descriptor chain with the appropriate number of entries and then sets the TXQUEUED bit in the COMMAND register.

Under no circumstances should the software driver set up a circular transmit queue with a single transmit descriptor pointing to itself. A minimum of two descriptors is required for proper EPIC operation.

Descriptor entries describe the location of transmit data in host memory. Data for a single transmit frame may not always be in a contiguous block in host memory. The LAN83C171 allows the software to specify multiple data buffers for each frame. Each frame may be queued in one of two ways, both of which may be used in the same descriptor chain.

### Direct Queuing Method

Descriptors point directly to the transmit data buffers.

One or more descriptors may be used to point to a single frame. All descriptors must have the FRAGLIST control bit set to 0. The first descriptor must contain the transmit length for the frame. The last descriptor for the frame must have the LASTDESCR bit set to 1 and contain the desired values for the TXIAF and NOCRC control bits. When the TXQUEUED bit is set, the transmit DMA will read the 4 dword descriptor from the location in host memory pointed to by its Current Descriptor Address register. If the ownership bit in the descriptor is equal to 1 then the LAN83C171 will accept the descriptor and update its Current Descriptor Address register with the value in the Next Descriptor Address field. Otherwise, the TXQUEUED bit will be cleared (and the transmit queue empty (TQE) interrupt set) and the Current Descriptor Address register will not be changed. The Transmit Length field in the first descriptor will always contain the number of bytes to be transmitted on the network, and not necessarily the number of bytes in the transmit buffers. The transmit DMA will begin copying data from the location in host memory specified by the Buffer Address field in the first descriptor. It will compare the transmit byte count to the Data Length field, and copy the lesser number of bytes into the local transmit RAM. If early transmit is enabled, the LAN83C171 will automatically initiate transmission on the network when the number of bytes specified in the Early Transmit Threshold register have been loaded into the transmit buffer.

If the transmit byte count is less than the Data Length field, or the LASTDESCR bit is set, then the frame copy is complete after the buffer has been read. The LAN83C171 will initiate transmission on the network if it has not already done so.

If the Data Length field is less than the transmit byte count and the LASTDESCR bit is not set, the LAN83C171 will attempt to read another descriptor. The transmit DMA will

proceed as before, however this time it will not read the Transmit Length field, but instead use the remaining number of bytes in its transmit byte counter (original byte count minus bytes already copied). This process will continue until a descriptor is read with the LASTDESCR bit set or the transmit byte count reaches zero. If LASTDESCR is set and the total number of bytes copied do not add up to the transmit byte count, the transmit MTU will pad the frame with random data after copying all of the valid data out of the transmit RAM. The CSMA/CD state machine will not append the automatically generated CRC to the frame if NOCRC is set in the last descriptor for the frame.

After the LAN83C171 has initiated the first transmission, it will check to see if there are any more frames in the transmit queue. If the software does not have another frame ready for transmission, the ownership bit in the next descriptor must be 0. If the ownership bit is 0, the LAN83C171 will clear TXQUEUED and set the transmit queue empty interrupt. If the ownership bit is 1, the LAN83C171 will begin copying the next frame into the local transmit RAM. The DMA will continue copying transmit buffers until the frame has been completely loaded into the transmit RAM or the first transmission has completed. If the copy completes while the first transmission is still in progress, the LAN83C171 will stop and wait. When the transmission is finished, the LAN83C171 will post the status into the first descriptor for that frame and immediately initiate the second transmission. If the transmission completes before the copy is done, the LAN83C171 will pause after the current transmit buffer has been copied and post the status from the first frame. If the early transmit threshold has already been exceeded then the second transmission will be initiated immediately. The transmit DMA will then continue by reading the next descriptor for the copy in progress.

When the transmit status is posted, the ownership bit will be written as 0 to indicate that the host now owns that descriptor again. The

Transmit Length field will not be overwritten. If TXIAF is true in the last descriptor for the frame, then the transmit complete (TXC) interrupt will be set. When there are no frames left in the queue and the last transmission has completed on the network, the transmit DMA will set the transmit chain complete (TCC) interrupt and return to its idle state.

Note: The difference between the TQE and the TCC interrupt is that the TQE interrupt is set only when there are no frames left in the queue. The TCC interrupt is set only when there are no frames left in the queue AND the last frame in the transmit queue buffer has been transmitted. Hence the TQE interrupt is set first and may or may not be followed by a TCC interrupt.

### **Fragment List Method**

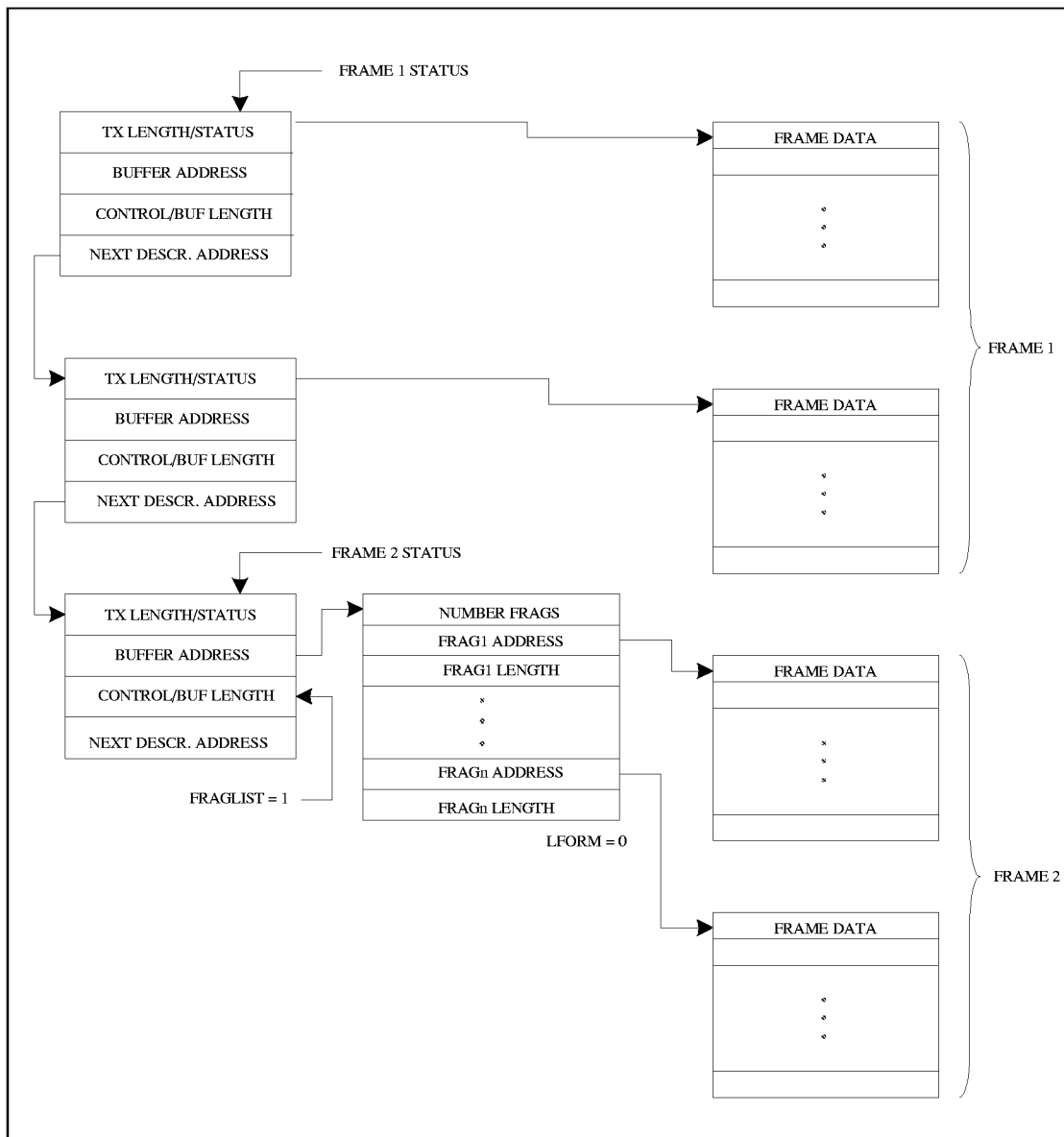
Descriptor points to a fragment list.

This method of queuing a transmit frame is much like the first method, except that each frame is always specified by one descriptor which points to a list of buffers (fragment list) instead of the buffers themselves. The FRAGLIST bit in the descriptor must be set to 1 and the LFFORM bit must properly indicate the format of the fragment list. The first entry

in the fragment list tells how many data buffers (fragments) are listed. Up to 63 fragments are allowed. The remaining entries specify the starting address and length of each buffer.

As in the direct queuing method, the transmit DMA will copy fragments one at a time into the local buffer until Transmit Length bytes have been copied or all of the fragments have been read. If early transmit is enabled, transmission will be initiated when enough bytes have been copied to meet the early transmit threshold. Otherwise transmission will be initiated when the entire copy is complete.

When more than one frame is queued, the transmit DMA will begin copying a second frame while the first is being transmitted. It will continue copying fragments until the entire frame is loaded or the first transmission has completed. If the copy completes while the first transmission is still in progress, the LAN83C171 will stop and wait. When the transmission is finished, the LAN83C171 will post the status into the first descriptor and immediately initiate the second transmission. If the transmission completes before the copy is done, the LAN83C171 will pause between fragments to post the status and then resume the copy. If the early transmit threshold has already been exceeded then the second transmission will be initiated immediately. Figure 3 shows a drawing of the Transmit Buffer Structure.



**FIGURE 3 - TRANSMIT BUFFER STRUCTURE**

The software may add transmit frames to the queue at any time. If the transmit process is already running (TXQUEUED may still be set)

then all of the descriptor and fragment list fields for the new frame must be valid BEFORE the ownership bit in the first descriptor is set. After

the descriptors are written, the TXQUEUED bit should be set.

TXQUEUED can be written regardless of completion status and will ensure that the latest frame is transmitted. If the LAN83C171 reaches the end of the transmit queue before the new frame has been added, a transmit chain complete interrupt is generated for the old portion of the queue and another transmit chain complete interrupt will be generated when the added portion completes.

### **Interrupting Transmit Chain**

The host may interrupt the transmit chain before all frames have been transmitted by setting the STOP\_TDMA bit in the command register. Setting this bit forces TXQUEUED to 0. The transmit DMA will finish copying any frame that it has already begun, and transmit all frames that have been loaded into the transmit ram. After the transmit DMA has posted the status for the last frame, it will set the transmit chain complete interrupt and return to its idle state (exactly as if the next frame in the queue was owned by the host). If the DMA reads a descriptor owned by the host while a copy is still in progress, it will set the transmit queued empty interrupt and wait for the descriptor to be re-queued. It will not return to the idle state until the copy is completed.

### **Transmit Buffer Full**

Whenever the local transmit RAM becomes full, the transmit DMA will wait until more space is available before loading any more data. Space is freed up as the transmit MTU reads data from the local RAM and updates its pointers. In some cases, the transmit MTU will leave its pointers at the beginning of a frame until it knows that the transmission will not have to be retried. Automatic retries can occur due to collisions or early transmit underruns.

### **Transmit Underrun**

A transmit underrun occurs in early transmit mode when the transmit DMA cannot keep up with transmission on the network. Data must be read from the local RAM before it is available. Usually, when an underrun occurs, the transmit MTU will generate a transmit underrun (TXU) interrupt and update its transmit status register. The transmit DMA will continue to operate as though nothing has happened. The software driver will be allowed to read the transmit status value from TXSTAT and set the "transmit underrun go" (TXUGO) bit to tell the MTU to retry the frame. The MTU will re-transmit the entire frame out of the local transmit ram. When transmission has completed successfully, the DMA will post the transmit status for the retry to the descriptor for that frame. Operation will continue as it normally would for a non-underrun situation.

### **Exception to Underrun ReTransmission**

The only exception to this behavior is when the transmit MTU cannot automatically retry an underrun frame. This happens when the frame size is larger than the transmit RAM (1.5 Kbytes) and the transmit DMA has overwritten the beginning of the frame before the underrun occurs. If such an event occurs, the transmit DMA will abort the copy and reset its pointers to the first descriptor for that frame. The DMA will clear the TXQUEUED bit and return to its idle state. Transmit queue empty and transmit chain complete interrupts will be generated along with the transmit underrun interrupt. The software driver must set TXUGO to reset the transmit MTU and then set TXQUEUED if it wants to retry the frame. The frame will be re-copied from scratch out of host memory when TXQUEUED is set.

### **Maximum Transmit Size and Burst Rate**

The transmit DMA supports frame sizes up to 64 Kbytes. The maximum size for a single data

buffer (fragment) is also 64 Kbytes. The transmit DMA will run at the maximum PCI data rate of 132 MByte/sec when the target memory system supports zero wait state reads. The transmit DMA will burst as many words as it can before having to relinquish the PCI bus. It is capable of bursting data continuously with no wait states (even when a transmission is active on the network) until the transmit RAM becomes full. The transmit DMA, however, will most likely lose possession of the PCI bus several times before it can fill the entire 1.5 Kbyte transmit buffer.

## RECEIVE DMA

The software driver initializes the receive process by writing the receive control register, interrupt mask register and general control register. The software must also program the PCI Receive Current Descriptor Address Register (PRCDAR) with the address in host memory where the first receive descriptor will be located.

To allow packet receptions, the software driver programs the receive descriptor chain and then sets the RXQUEUED and START\_RX bits in the COMMAND register. Setting START\_RX brings the CSMA/CD receiver online. The receive DMA is enabled by setting RXQUEUED. The software driver should set RXQUEUED before or simultaneous to bringing the receiver online so that the receiver does not overflow the local buffer while waiting for a descriptor to be queued. The first descriptor must be valid before the RXQUEUED bit is set. The first descriptor will be read as soon as it is queued, even if no receptions have occurred on the network.

The receive lookahead method offers maximum performance in most cases.

## Free Buffer Pool Method

In this mode the software driver pre-allocates a pool of free buffers for frames received by the LAN83C171. The ONECOPY bit in the general control register must be set so that the each frame may be copied into the buffer pool without host intervention. The descriptors for the free buffer pool may point directly to the buffers, or point to a fragment list which in turn specifies the buffers.

When the RXQUEUED bit is set, the receive DMA will attempt to read the first descriptor from the address pointed to by its Current Descriptor Address register. If the ownership bit is 0, the RXQUEUED bit will be cleared (and the receive queue empty (RQE) interrupt set) and the Current Descriptor Address register will not be changed. If the ownership bit is equal to 1, the LAN83C171 will accept the descriptor and update its Current Descriptor Address register with the value in the Next Descriptor Address field. The LAN83C171 will save the descriptor information until a frame is received. If the fraglist control bit is also 1, then the receive DMA will read and save the address pointer and data length for the first buffer in the fragment list. The offset field in the descriptor (see buffer length field) should be set to zero, otherwise the copy will not begin at the start of the frame. The fragment list format for the receive DMA is identical to the format for the transmit DMA.

As soon as a frame is received, the LAN83C171 will begin copying it from the local receive buffer into the allocated buffer in host memory. If early receive is enabled, the LAN83C171 can begin the copy while reception is still in progress. The receive DMA always monitors the local buffer contents so that a receive underflow can never occur. As soon as the receive DMA has copied the number of bytes in the PCI Receive Copy Threshold register, it will set the receive copy threshold (RCT) interrupt. When the receive DMA has copied the entire packet from the local RAM into host memory, it will

post the receive status into the first descriptor for the frame and set the receive copy complete (RCC) interrupt. The DMA will read the next descriptor and, if owned by the NIC, check to see if there are any more frames to copy out of the local RAM. If the receive DMA fills the first host buffer before the entire frame has been copied, it will read the next descriptor or fragment list entry to find more buffer space. This process will continue until the entire frame has been copied. If the DMA reads a descriptor with the ownership bit set to 0, it will clear the RXQUEUED bit (and set the receive queue empty interrupt) and wait for a new descriptor to be queued. In fragment list mode, the receive DMA always expects the fragment list to contain enough buffer space for the entire frame.

If all the buffers in the fragment list are filled before the copy is finished, the DMA will abort the copy and set the fragment list error bit in the PRSTAT register. The DMA receive status will be posted to the descriptor for that frame and the RXQUEUED bit will be cleared. If early receive is enabled, the network portion of the receive status may not be valid yet, as indicated by the RSV bit posted in the status. The software driver may poll the RSV bit in the interrupt status register, and when it returns a 1 read the receive status from PRSTAT. The software may attempt to re-copy the frame by setting the RXQUEUED bit again, or may discard the frame by setting the NEXTFRAME bit before or simultaneous to setting RXQUEUED. If RXQUEUED is set after or along with NEXTFRAME, the DMA will begin to copy the next frame (if any) in the receive buffer.

**Note:** The DMA rounds the number of bytes copied up to the nearest dword. If the receive buffer does not start on a dword boundary, the number of bytes in the receive buffer may be slightly less (up to 3 bytes) than the receive copy threshold when the interrupt is generated.

## **Adding Receive Buffers to the Pool**

The software driver adds buffers to the pool by writing the appropriate descriptors and setting their ownership bit to 1. If the receive DMA has stopped (RXQUEUED is cleared), the software must set the RXQUEUED bit to queue the new descriptors. The RXQUEUED bit may be set at any time, even if the receive DMA is still active.

## **Receive Lookahead Method**

When this buffering method is used, the LAN83C171 first copies only the header of a frame into host memory, and then waits for a queue from the software driver before copying the rest of the frame. The software usually specifies the final destination of the frame data with a fragment list. The advantage to this buffering method is that the LAN83C171 may copy frame data to its final destination instead of a temporary buffer space, so the software driver is not required to re-copy the data from one host memory location to another.

In receive lookahead mode, frames are usually copied into the host memory one at a time, and a handshake is performed between the software driver and the LAN83C171 during each frame. The handshake is performed using the RXQUEUED and NEXTFRAME bits in the COMMAND register, the receive copy complete (RCC), and header copy complete (HCC) interrupts. The control bits in the receive descriptors are also used to direct the receive DMA.

The software driver begins by setting up a buffer for the header of the first frame and setting RXQUEUED. The HEADER control bit in the descriptor should be set and the FRAGLIST bit should be cleared. The buffer address pointer and length are specified directly in the descriptor. When a frame is received, the receive DMA begins copying the beginning of the frame into the header buffer until the buffer is full, or until the entire frame has been copied.

The copy may begin before the entire frame has been received if early receive is enabled. When the header copy is complete, the receive DMA status will then be posted to the descriptor for the header buffer, and the header copy complete interrupt will be set. If reception from the network has completed, the network portion of the posted status will be valid --and the RSV bit will be set to 1. In early receive mode, the receive DMA status may be posted before the network status for the frame is available, in which case the RSV bit in the descriptor will be set to 0. If the entire frame fits into the header buffer, the network receive status will always be posted with the frame. After a header copy, the receive DMA always clears the RXQUEUED bit (also setting the receive queue empty interrupt, which may be masked) and waits in the idle state for the software driver to queue a fragment list for the rest of the frame.

After examining the header data, the software driver may discard the frame or have it copied into host memory as many times as it would like. The software requests copies of the frame by programming descriptors (and fragments lists) and setting RXQUEUED without setting NEXTFRAME. The frame is copied exactly as it would be in the free buffer pool mode, with the exception that the offset field is used with fragment list copies. The software may not need all of the bytes at the beginning of the frame to be copied, so it may specify an offset into the frame where the copy should begin. The offset field shares a location in the descriptor with the buffer length field because the buffer length is not specified in a descriptor for a fragment list. The receive DMA copies the frame into host memory beginning from the byte number specified in the offset. If the offset field is not zero, the copy will not begin until the entire frame has been received from the network, even if early receive is enabled. This is so that the receive DMA does not copy invalid data if the offset is greater than the number of bytes that have been received so far. Usually,

the entire frame will have been received before the fragment list is available.

When the copy is finished, the receive status is posted and the receive copy complete interrupt is set. The receive DMA will then read the next descriptor, and if the ownership bit is set it will immediately begin to copy the same frame into host memory again. If the descriptor is owned by the host, RXQUEUED will be cleared (and receive queue empty interrupt set) and the receive DMA will wait in the idle state for another command. If the software driver wants another copy of the frame, it may queue another descriptor and set RXQUEUED without setting NEXTFRAME. This procedure will be repeated until the software chooses to go on to the next frame.

The software driver discards a frame by setting NEXTFRAME before or simultaneous to setting RXQUEUED. If RXQUEUED is set after or along with NEXTFRAME, the receive DMA will begin to copy the next frame (if any) in the receive buffer. The next descriptor queued should contain a header buffer for the next frame.

Occasionally, the software driver may want to discard a frame immediately after reading its header, but still read the receive status for that frame. If the valid network status is not posted in the descriptor, the software driver may read it from the PRSTAT register. The driver must set NEXTFRAME and RXQUEUED to discard the frame first, as described above. However, the next descriptor in the receive descriptor list must have the ownership bit cleared (host still owns descriptor). This allows the LAN83C171 to update the PRSTAT register without starting to copy the following frame. The software driver must poll the RQE (receive queue empty) interrupt to determine when the status is available. When the RQE interrupt is set, the driver may read the receive status from the PRSTAT register. The receive status valid bit in

the interrupt status register will not indicate when the receive status is available.

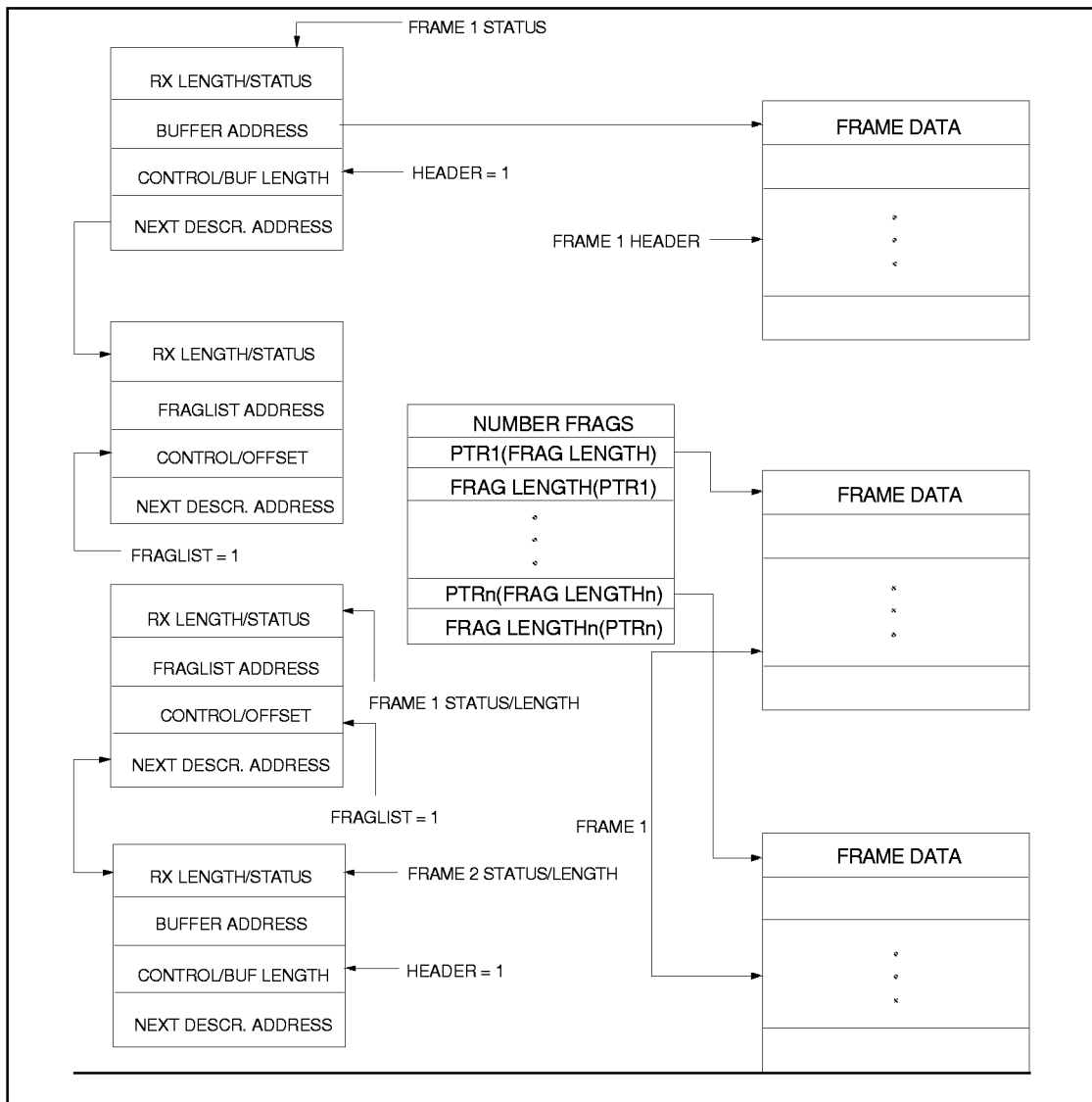
When the software driver wants only one more copy of the current frame, it does not have to wait for the copy to complete before setting NEXTFRAME. The software may set NEXTFRAME immediately after setting RXQUEUED (on the following I/O write) and begin to queue the header descriptor for the next frame. After the header descriptor is queued, the software may set RXQUEUED again to

guarantee that the header descriptor is recognized. When the DMA is finished copying the first frame, it will immediately read the next descriptor and may begin copying the next header without waiting for the software to respond to an interrupt.

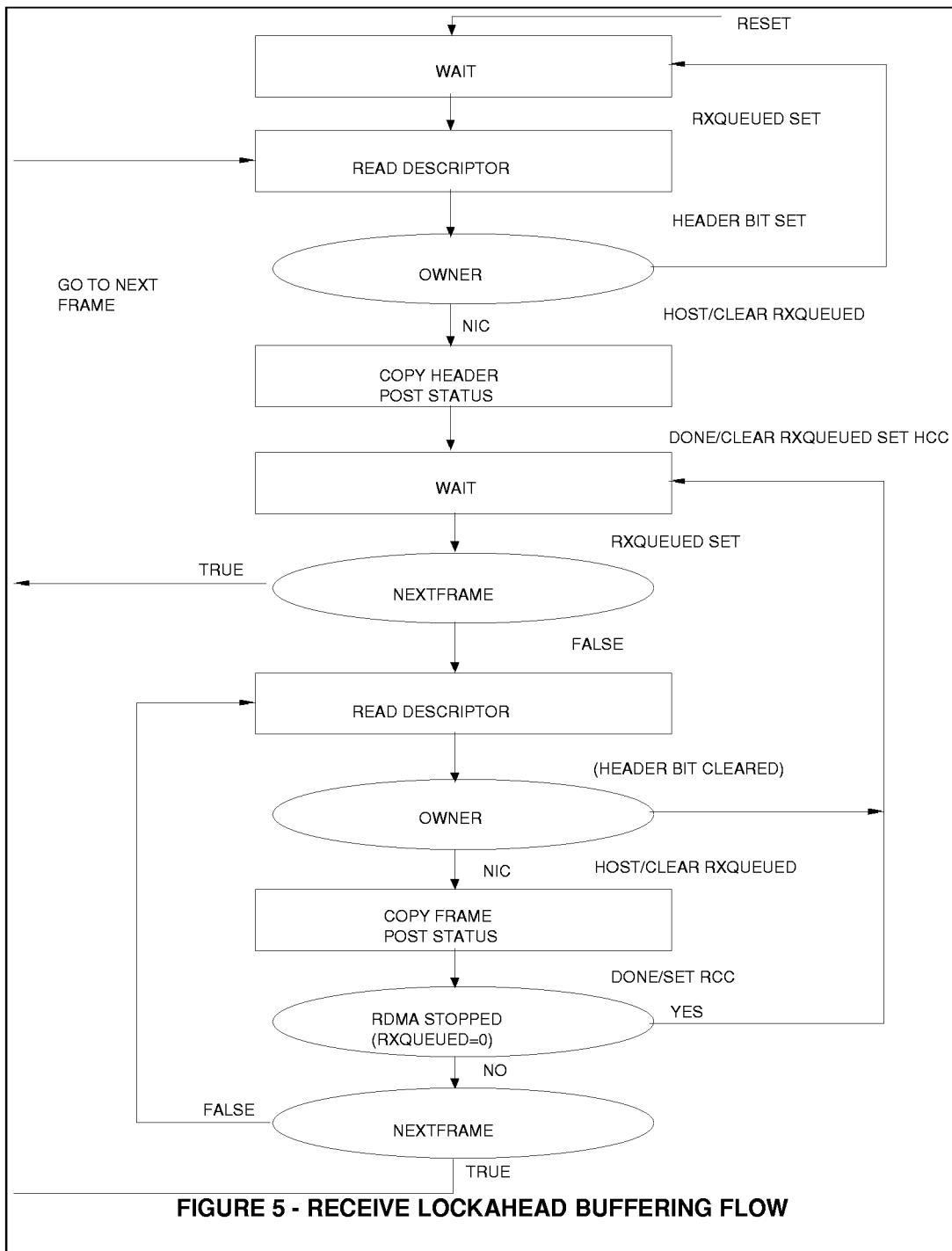
Note: Software must never set NEXTFRAME more than once per frame. NEXTFRAME may only be set when the copy is in progress or has already been completed.

Figure 4 shows an example of the Receive

Buffer Structure and also shows a flow diagram for the Receive Buffering Method.



**FIGURE 4 - RECEIVE BUFFER STRUCTURE**



**FIGURE 5 - RECEIVE LOCKAHEAD BUFFERING FLOW**

## Stopping the Receive DMA

The receive DMA may be halted by setting the STOP\_RDMA bit in the command register. Setting this bit forces RXQUEUED to 0. The CSMA/CD receiver should also be taken offline to prevent it from continuing to buffer receive frames. The receive DMA will attempt to complete any copy in progress. When finished, it will return to its idle state. When the CSMA/CD receiver is offline and has also returned to its idle state, the RXIDLE bit in the interrupt status register will become true (1). If the DMA reads a descriptor owned by the host before it completes its current copy, it will set the receive queued empty interrupt and return to the idle state. The DMA will continue the copy when more buffers are queued. The software driver can tell if a copy is still in progress or if there are any more frames in the local receive RAM by reading the RCIP and RBE bits in the interrupt status register.

The STOP\_RDMA bit can be set when the receive DMA has read and saved the information in a descriptor, but there are no frames in the local receive RAM. In this case, the receive DMA will reset its current descriptor pointer back to that descriptor and return to the idle state. When the RXQUEUED bit is set again, the DMA will be re-read the descriptor.

## Maximum Receive Size and Burst Rate

The receive DMA supports frame sizes up to 64 Kbytes. The maximum size for a single data buffer (fragment) is also 64 Kbytes. The receive DMA will run at the maximum PCI data rate of 132 Mbps when the target memory system supports zero wait state writes. DMA bursts, at this rate, will run for a limited number of dwords. The length of each burst is dependent on the FIFO threshold level and access to the local

receive RAM. The receive DMA loads data into the receive burst FIFO at a maximum rate of 100Mb/s (when reception is not in progress) or 83 Mb/s (when reception is in progress). The receive DMA will automatically initiate a burst on the PCI bus whenever the FIFO reaches programmed threshold level. The receive DMA will continue to load data into the FIFO while it is being emptied onto the PCI bus. The burst will continue until the FIFO is empty or the receive DMA loses control of the PCI bus (to the internal transmit DMA or to another PCI master). Another burst will begin when the FIFO reaches the threshold level again, or when the last of the data for the current copy has been loaded into the FIFO. The PCI bus will be requested immediately if the receive DMA loses possession of the bus while the FIFO is above the threshold level.

| THR_SEL<br>[1] | THR_SEL [0] | THRESHOLD<br>LEVEL  |
|----------------|-------------|---------------------|
| 0              | 0           | 1/4 Full (32 Bytes) |
| 0              | 1           | 1/2 Full (64 Bytes) |
| 1              | 0           | 3/4 Full (96 Bytes) |
| 1              | 1           | Full (128 Bytes)    |

A lower threshold allows the LAN83C171 to begin moving data on the PCI bus sooner, while a higher threshold may allow longer bursts. A higher threshold level will not result in longer data bursts if the receive FIFO never reaches the empty level (due to latencies on the PCI bus). The default (reset) threshold level is 1/2 full.

## **MAC OPERATION**

The LAN83C171 is compliant with the 802.3 standard CSMA/CD protocol for 10 or 100Mb/s Ethernet networks.

### **MAC TRANSMITTER**

The LAN83C171 CSMA/CD transmitter is capable of generating network data at rates of 10 and 100Mb/s. It supports current implementations of 10Mb/s physical layer devices, and the 802.3u Media Independent Interface (MII) for 10 and 100Mb/s.

#### **Basic Function**

The transmitter generates serial and nibble wide data streams at 10 or 100Mb/s. It forms a proper preamble and SFD field at the beginning of each packet. The frame data is then shifted serially or by nibbles from an internal transmit buffer to the physical layer. The transmitter completes the packet by computing and appending the CRC field. During packet transmission, the transmitter monitors the network for collisions and retransmits frames after a random backoff time when necessary. The transmitter maintains the transmit statistics and generates status information on each attempted transmission. Optional operating modes can be selected by programming the transmit configuration register.

#### **Preamble Generation**

At the beginning of each packet, the transmitter generates 56 bits of preamble (an alternating '1010' pattern). Following the preamble, a Start of Frame Delimiter (a '10101011' sequence) is transmitted.

## **Transmit Serializer**

The transmit serializer converts 8 bits of parallel data from the transmit buffer to serial or nibble wide data. The mode of operation is selected by the MII configuration register. In serial mode, the transmit data is shifted out of the TXD[0] pin synchronous to transmit clock. Data is shifted out least significant bit (LSB) first. In nibble mode, data is shifted synchronous to the transmit clock at a one nibble per clock rate. The data is transmitted least significant nibble first on pins TXD[3-0]. The LSB is transmitted on pin TXD[0].

### **CRC Generator**

The transmitter calculates the CRC and appends it to each packet. CRC data is clocked out most significant bit (MSB) first. A packet can be transmitted without an attached CRC field by programming the transmit descriptor. This is provided for bridging applications in which the original checksum must remain attached to the packet until the final destination.

### **Transmit Protocol FSM**

The transmit protocol FSM controls the transmission of packets by monitoring collisions, deferring to active carriers and collisions, and initiating backoff when needed.

## Interframe Gap and Deference

Deference is initiated when both CRS and COL have terminated at the end of a packet. The transmitter deference logic initiates a 2-part timer at the end of network activity. While this timer is running, no frame transmission will be initiated. The first part of the timer (interFrame SpacingPart1) is used to observe the network for transmission activity by other stations. If this station is transmitting, carrier is sensed, or collision is detected during this part of the timer, the timer will be reset to zero and held there until the termination of line activity. When the first part of the timer elapses, line activity is no longer observed and the timer runs to completion.

If any frame is queued up for transmission at the moment of timer expiration, transmission will be initiated regardless of line activity.

The combination of interFrame SpacingPart1 and interFrame SpacingPart2 makes up the Inter-Frame Gap (IFG) as defined by the 802.3 specification.

## Collision Handling Logic

When collision is detected by the transmitter section during the first slot timer of an active transmission, the transmission terminates after completion of the preamble and the jam sequence. The jam sequence is 32 bits of logic '1's. If collision is detected after the slot time is passed, the transmission will be aborted after the jam sequence. An out of window collision interrupt will be generated and the collision count status will be retained in the transmit status register for software collection. After the software has responded to the interrupt and re-enabled transmissions, the transmit status will be cleared and the packet will be automatically retransmitted.

## Timers and Counters/Slot Timer

During transmit, the slot timer starts counting once the receiver recognizes that a carrier is present at the start of a returning preamble. When backing off, the slot timer starts with the end of transmit enable (TX\_EN) for the collided frame and is not reset by any other incoming frames. The slot timer is programmable by the transmit control register. The default slot time is 512 bit times.

## Backoff Timer

After a transmission is terminated because of a collision, a retransmission is attempted. The backoff time is determined by the truncated binary exponential backoff algorithm. This algorithm is:

Draw a random integer  $r$  such that  $r \cdot 0 \leq r < 2^k$

$k$  is the number of retries already on this transmission. The value  $k$  is initialized to 0.

The required backoff time is ' $r$ ' number of slot times. After the backoff time has been completed, normal transmission deferral begins.

The backoff timer is a 12 bit counter that is initialized to a random number when an attempted transmission results in a collision. The counter decrements once per slot time until it reaches zero. The transmit protocol FSM utilizes this timer to insert a variable amount of delay ahead of its attempt to retransmit the frame.

## Collision Counter

Prior to the first attempt at each frame transmission, the collision counter is initialized to 0. Each attempted transmission of the frame resulting in a collision causes the collision counter to increment. If the maximum number of collisions (16) is reached before a successful attempt to transmit the frame, the frame

transmission is aborted. An interrupt is generated for an aborted frame indicating transmission complete, and the collision count value in the transmit status register is 16.

### **Heartbeat Detection**

When the transmitter is configured in serial mode, after each transmission, the transmit logic opens a window 3.6µsec long during which it looks for a pulse on the COL pin. This pulse is normally generated by the MAU and is received through the MII interface. If the pulse is received, the CDH status bit of the transmit status register is cleared. If no pulse is received, the CDH bit is set.

### **MAC RECEIVER**

The LAN83C171 CSMA/CD receiver is capable of operating with network data rates of 10 and 100Mb/s. It supports current implementations of 10Mb/s physical layer devices, and the 802.3u Media Independent Interface for 10 and 100Mb/s.

### **Basic Function**

The receiver processes serial or nibble wide data streams at data rates of 10Mb/s or 100Mb/s. The receiver detects start of frame, provides destination address recognition and filtering, transfers recognized frames to memory, and provides error detection and reporting.

### **Interface to Physical Layers**

The receiver interfaces to the physical layer in serial or parallel mode. When in the serial mode, data is transferred serially on the RXD[0] pin synchronous to the falling edge of the receive data clock (RX\_CLK). RX\_CLK is a 10MHz clock signal recovered by the physical layer device from the data stream. The CRS and COL signals provide carrier sense and collision detect respectively.

In parallel mode, the physical layer device transfers data to the LAN83C171 four bits at a time on the RXD[3-0] data bus. The data is transferred synchronously to the falling edge of RXC. The signal Receive Data Valid (RX\_DV) informs the MAC of the RXD bus status. The physical layer can also notify the LAN83C171 of invalid data on the medium with the Receive Error signal (RX\_ER). Selecting the receiver interface mode is performed by programming the MII Configuration register.

### **Packet Reception/Serial Mode**

After detection of carrier, serial bits received on RXD[0] are synchronized to the rising edge of RX\_CLK. Each bit is shifted through an 8 bit shift register scanning for a Start of Frame Delimiter (SFD) pattern of '10101011' received from left to right. Following detection of SFD, all bits are byte aligned in the serial to parallel shift register. Bits are received from least significant bit to most significant bit within the byte. Data from the shift register is transferred to the receive FIFO where it waits for the receive DMA to transfer it into local memory. The receive process continues while CRS or COL are active.

### **Parallel Mode**

Packet reception begins with the first nibble after detecting RX\_DV active. Nibbles transferred on RXD[3-0] are synchronized to the rising edge of RX\_CLK and shifted into a 2-nibble shift register. RXD[0] is the least significant bit (LSB). SFD is detected when the shift register contains the value '10101011' from LSB to MSB. The preamble and SFD pattern received from the PHY device is required to be nibble aligned. Bytes are aligned in the 2-nibble shift register after detection of SFD. Each byte is transferred to the receive FIFO. Packet reception continues while RXDV is active and ends with the nibble preceding the falling edge of RX\_DV. While RX\_DV is de-asserted, the value of RXD[3-0] has no effect on the MAC.

## Error Detection

The receiver computes the CRC of incoming frames for all data following the detection of SFD in both parallel and serial mode until the end of frame, including the CRC field.

Computation stops after the reception of the last whole byte following loss of carrier in serial mode or the transition of RX\_DV from active to inactive in parallel mode. The final value of the CRC must be "C704DD7B" for the packet to be validated. The CRC polynomial used is AUTODIN II ( $X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$ ).

In addition, in parallel mode, the receive error signal, RXER, forces a CRC error when it is asserted while RX\_DV is active.

## Address Recognition

The receiver is capable of recognizing individual, multicast, and broadcast addresses. It can also be programmed to operate in promiscuous mode and receive all frames regardless of address. In all cases, address recognition begins with the first byte following SFD and ends with the sixth byte after SFD. Individual destination addresses are compared against a 6 byte register station address. Multicast addresses are recognized by taking a 6 bit snapshot of the partially computed CRC as the end of the destination address field passes through the CRC checker. If the address has the multicast bit set, the 6 bits are used as a hashed index to a 64 bit Multicast Filter table. If reception of multicast frames has been enabled and if the 6 bit hash index points to a bit in the table that has been set, the multicast frame will be recognized. Broadcast frames are received when the broadcast enable bit is set and the destination address specifies a broadcast frame, or when the hashed bit in the Multicast Filter table has been set.

Reception of all multicast and broadcast frames can be achieved by setting all bits in the Multicast filter table and enabling reception of multicast frames. If the address is not recognized by any of the above means, then the frame will be ignored.

When an address is recognized, the entire frame will be saved into local memory.

## Frame Processing

Frame processing begins following the detection of SFD and continues until the last bit or nibble of the frame has been received. Frame processing counts the number of bytes in the receive frame, transfers data to the receive FIFO, checks for errors in both size and data, posts status, generates interrupts, and counts events. Frame processing can be controlled by the receive control register to allow flexible control of frame reception.

## Receive Byte Count

The receive byte counter begins counting with the first byte of SFD and counts all bytes of the frame until the end of frame is detected or an error condition causes the frame processing to be aborted. The counter filters runt frames by comparing the current byte count value to the slot time programmed in the Transmit Control register. The frame is considered a runt until the byte count exceeds the slot time value. Runt frames are not received under default conditions. Reception of runt frames can be enabled by setting the receive runts bit in the receive control register.

## Data Transfer

Receive data is stored temporarily in a 8-dword receive FIFO. Data begins to be stored in the FIFO after detection of SFD. If the destination address is not recognized, data stops being transferred into the FIFO and the FIFO is reset. When the FIFO level reaches 6-dwords, a burst

request to the local receive memory is made. The return of acknowledge is guaranteed to prevent the receive FIFO from overflowing. The data path to the receive local memory is 32 bits.

If the receive local memory becomes full during reception of a frame, the frame is aborted. The host is notified of the condition with an overflow interrupt. Additionally, the missed packet counter is incremented for each frame which could not be received due to the overflow condition.

### **Monitor Mode**

The data transfer process can be inhibited by operating in monitor mode. This mode checks validity of incoming frames and maintains error statistics, but does not store the frame in memory. Frames, which would otherwise have been accepted, cause the Missed Packet counter (MPCNT) to increment upon completion of the frame.

MPCNT register records the number of overflow packets when the EPIC is operating in normal mode. It's function changes when in monitor mode and counts the number of packets that were meant for this node but not saved.

### **Error Checking**

Received frames are checked for CRC and alignment errors. If the CRC of a received frame is incorrect, a CRC error is indicated in the status register and the CRC error counter is incremented. Reception of the frame is aborted unless the receiver has been programmed to receive errored packets. If the frame does not terminate on a byte boundary and the CRC is

incorrect, then an alignment error is also indicated in the status register. When this occurs, the alignment error counter will be incremented. A receive error interrupt is generated when a CRC error is detected and monitor mode is not set.

The receive control register can be programmed to enable long frame checking (frames longer than 1518 bytes). When a long frame is detected the CRC and alignment counters are not incremented.

### **Status**

A status register is updated at the completion of each frame whether it completed normally or aborted in error. The status register holds important information about the frame until it is transferred with the packet data to the receive local memory. If the frame data is not being saved due to an error or monitor mode, then the contents of the status register will be lost after the completion of the following packet. A description of the status register contents is located in the register definition section of this data sheet.

### **Event Counters**

Three event counters record CRC errors, alignment errors, and missed packets. The counters are all 8 bits wide and count from zero to 255. At 255 the counters stop until they have been read by the host. The counters are self clearing after the read. The counters generate a shared interrupt when any one of them reaches a count of 192.

The counter is also incremented for receive local memory full errors. The missed packet counter is 8 bits wide and generates an interrupt when it reaches a count of 192.

## MII MANAGEMENT INTERFACE

The LAN83C171 supports the 802.3 specification for the MII Serial Management Interface. The interface consists of two pins: the clock, MDC; and the data, MDIO. The LAN83C171 PHY interface Register allows writing one bit at a time or the entire PHY register once. Writing an entire PHY register option permits lower PCI bus utilization. The LAN83C171 serializes this data and writes it to the PHY via MDC and MDIO.

## EEPROM INTERFACE

The LAN83C171 has a serial interface to an external EEPROM. The serial EEPROM contains the LAN Address for the adapter and a few bytes of configuration information. The LAN address and configuration information is automatically recalled from the first eight words (each word is 16 bits) of the 8 bit parallel EEPROM after reset. Access to the LAN83C171 is disabled during the initial EEPROM recall. Any attempted access results in a PCI target retry. The initial recall may be bypassed through a test "jumper".

### EEPROM Format

|                      |                    |
|----------------------|--------------------|
| Word 0, Bits 7 - 0:  | LAN Address Byte 0 |
| Word 0, Bits 15 - 8: | LAN Address Byte 1 |
| Word 1, Bits 7 - 0:  | LAN Address Byte 2 |
| Word 1, Bits 15 - 8: | LAN Address Byte 3 |
| Word 2, Bits 7 - 0:  | LAN Address Byte 4 |
| Word 2, Bits 15 - 8: | LAN Address Byte 5 |
| Word 3, Bits 7 - 0:  | Board ID           |
| Word 3, Bits 15 - 8: | Checksum           |

|                      |                                        |
|----------------------|----------------------------------------|
| Word 4, Bits 5 - 0:  | Non-Volatile Control Register Contents |
| Word 5, Bits 7 - 0:  | PCI Minimum Grant Desired Setting      |
| Word 5, Bits 15 - 8: | PCI Maximum Latency Desired Setting    |
| Word 6, Bits 15 - 0: | Subsystem Vendor ID                    |
| Word 7, Bits 14 - 0: | Subsystem ID                           |

The remaining words in the EEPROM are available to software for general purpose non-volatile storage. The LAN83C171 can operate with four different size EEPROMs. An external jumper selects which serial EEPROM is installed. After the initial recall, the serial EEPROM may be accessed through the EEPROM control register. The inputs of the EEPROM can be controlled directly through three bits in the register, and the EEPROM size and data can be read through the register. Each time the register is written, there is a delay before the read data is valid and/or the register may be written again. The EERDY signal indicates that this delay has expired (EERDY will stay low for 1280-2560ns after the register is written). The programming sequence for the EEPROM may be found in the manufacturer's data book. Because the EEPROM clock and data input are multiplexed with the LAN83C171 memory address bus, software must set the EEPROM enable signal and wait for EERDY before the programming sequence may be initiated. The EEPROM enable bit must remain set until the entire programming sequence has completed and EERDY has become true.

## JUMPER OPTIONS (EEPROM/RAM)

There are several operational modes in the LAN83C171 which are selected by "jumpers" at power-up reset. Actual jumpers do not need to be installed on the board. The options can be set by external pull-up or pull-down resistors at manufacturing time. Pins MD[4:0] are used to make the jumper selections. The pins are latched shortly after nRST goes inactive. The jumpers are not sampled after soft reset. A pull-down resistor sets the jumper value to 0, and a pull-up sets it to 1. The jumpers are defined as follows:

**MD[5] AUXILIARY POWER PRESENT** - When set to 1, indicates the presence of auxiliary power to support the monitoring of the network for wake-up events upon the loss of PCI Bus power.

**MD[4] EEPROM TEST** - When set to 1, the EEPROM clock is sped up for test purposes.

**MD[3] BYPASS EEPROM RECALL** - When set to 1, the EEPROM recall is not performed after power-up reset. Used for test purposes only.

**MD[2] nBIG EEPROM** - Indicates the size of the serial EEPROM. 0 = 128x16 or 256x16, 1 = 16x16 or 64x16.

**MD[1:0] EXTERNAL BUFFER SIZE SELECT** - Indicates the size of the external receive RAM according to the following table:

| MD [1] | MD [0] | BUFFER SIZE |
|--------|--------|-------------|
| 0      | 0      | 0 Kbytes    |
| 0      | 1      | 16 Kbytes   |
| 1      | 0      | 32 Kbytes   |
| 1      | 1      | 128 Kbytes  |

## Advanced Configuration and Power Interface (ACPI) Support

The EPIC incorporates all of the requirements and most of the optional features needed to comply with the network device class and PCI bus class sub-specifications of the ACPI specification. The provided power management capabilities within the EPIC can be determined by accessing the "Power Management Register Block" located in the EPIC PCI configuration address space. The Power Management Register Block also provides an interface to set or query the chip power state and to enable wake-up notification.

### Wake-Up Events and Notification

The EPIC can be enabled to provide wake-up notification upon the occurrence of a power management event (PME). The EPIC supports the detection of a Magic Packet and the occurrence of a network link-down event as PMEs. The detection of a Magic Packet can be enabled to generate a wake-up notification by setting the Magic Packet Enable bit in the General Control Register. Likewise, the occurrence of a link-down event can be enabled to generate a wake-up notification by setting the appropriate enable bit in the PHY's interrupt mask register. The monitoring of wake-up events will only occur when the EPIC is in one of the low power modes.

The software driver can specify if the EPIC notifies the OS of the occurrence of a PME via the Power Management Control and Status Register (bits 23 through 0/PMCSR - address E0h Configuration Address Space). If an enabled PME has occurred (bit 15/PMCSR set) and the EPIC is enabled to notify the OS (bit 8/PMCSR set), the EPIC will assert the signal nPME. (Note: nPME can be asserted and deasserted asynchronously to the PCI clock). The EPIC will deassert nPME when the driver either clears the PME status bit (bit 15) or disables the notification of a PME (bit 8) in the

PMCSR. The PME status and enable bits in the PMCSR will not be altered by the assertion of PCI reset. The bits will, however, be cleared at initial power-up by the EPIC as described below. Therefore, the EPIC will maintain the bits' states even during the loss of PCI sourced power to the chip (assuming that the EPIC is receiving power from an alternate source).

Note: The EPIC will not drive nPME to ground when not being asserted by the chip and the PCI sourced power to the EPIC has been removed.

## EPIC Power States

The software driver can place the EPIC into the fully operational mode D0 or one of the low power modes D1, D2, or D3 via the PMCSR. The effects of the assertion of the PCI Bus reset on the controller's power state will be selectable as described below. The EPIC power state will, however, be set to D0 at initial power-up. The EPIC will maintain a state even during the loss of PCI Bus sourced power to the EPIC assuming the EPIC is receiving power from an alternate source. The EPIC power states are:

|     | MODE    | DESCRIPTION                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|-----|---------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| i   | D0      | <ul style="list-style-type: none"> <li>Fully operational mode</li> <li>Will not scan the network for a Magic Packet</li> <li>If nPME was asserted in one of the other power states prior to transitioning to power state D0, nPME will remain asserted during power state D0 until either bit 15 and/or bit 8 within the PMCSR is cleared</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| ii  | D1      | <ul style="list-style-type: none"> <li>Same as D3(Hot)</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| iii | D2      | <ul style="list-style-type: none"> <li>Same as D3(Hot)</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| iv  | D3(Hot) | <ul style="list-style-type: none"> <li>Low power mode (PCI power present)</li> <li>Will assert the Power Down bit (bit 3) in the General Control Register (address 0Ch - EPIC I/O address space). Setting this bit will gate-off the controller's internal system clock</li> <li>If the PCI clock is still active, then PCI configuration space transactions will be possible</li> <li>Access to the EPIC I/O and memory address spaces will not be possible</li> <li>EPIC PCI-Bus-Master capabilities will be disabled</li> <li>EPIC PCI interrupt capabilities will be disabled</li> <li>Will scan the network for a Magic Packet</li> <li>Will maintain internal state intact</li> <li>Additional reductions in power consumption may be realized by gating-off the EPIC transmit clock. This can be accomplished by setting bits 15 and 14 to 11b in the MII Configuration Register (address 38h - EPIC I/O address space)</li> </ul> |

|    | MODE      | DESCRIPTION                                                                                                                                                                                                                                                           |
|----|-----------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| v  | D3(Cold1) | <ul style="list-style-type: none"> <li>• Auxiliary powered mode (no PCI sourced power)</li> <li>• Same as D3(Hot) with the following exception. As the PCI Bus clock will be disabled, access to the EPIC Configuration Address space will not be possible</li> </ul> |
| vi | D3(Cold2) | <ul style="list-style-type: none"> <li>• No power mode (auxiliary power not supplied)</li> <li>• Will not scan the network for a Magic Packet</li> <li>• No transactions possible on any bus</li> <li>• Internal state not preserved</li> </ul>                       |

Before placing the EPIC into one of the low power states, the driver shall ensure that there are no ongoing network receptions or transmissions in progress involving the EPIC (i.e. the EPIC is in an idle state). Otherwise, the corruption of such a transfer will be a distinct possibility.

### D3(Cold1) Software Driver Requirements

For the EPIC to scan the network for a Magic Packet, monitor the PHY's interrupt, and provide wake up notification during power state D3(Cold1), the driver is required to place the EPIC into low power state D3, set the Magic Packet enable bit (bit 18 of the General Control Register - address 0Ch - EPIC I/O address space) as desired, set the PHY's interrupt mask register as desired, and enable PME notification (bit 8/PMCSR) prior to the removal of PCI Bus power.

### Supporting Power Management Options

If the effects of the PCI Bus reset on the EPIC are to be avoided during power state D3(Cold1), bit 17 in the General Control Register (address 0Ch - EPIC I/O address space) may be set to a one(1) to make the EPIC ignore the transitions of the PCI Bus reset.

Also, the affect of gating-off the internal system clock during the low power states can be avoided by setting bit 16 in the General Control Register (address 0Ch - EPIC I/O address space) to a one(1) to make the EPIC restore its internal system clock.

PCI Bus reset assertion effect on the current power state can be selected by setting bits 21 and 20 in the General Control Register (address 0Ch - EPIC I/O address space) as follows:

| BIT 21 | BIT 20 | DESCRIPTION                                                                                                                                                                                             |
|--------|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0      | 0      | Initial Power Up State. The assertion of PCI Bus reset will immediately transition the EPIC power state to D0                                                                                           |
| 0      | 1      | The assertion of the PCI Bus reset will not have any effect on the chip power state. However, the transition from the PCI Bus reset being asserted to deasserted will change the chip power state to D0 |
| 1      | 0      | Transitions on the PCI Bus reset will not have any effect on the chip power state                                                                                                                       |
| 1      | 1      | Same as "10"                                                                                                                                                                                            |

### PME Generates a PCI Bus Interrupt

The EPIC provides the ability for an enabled PME assert the PCI Bus interrupt line (nINTA). To enable this feature during the low power modes, it will be necessary to set bit 19 to one(1) in the General Control Register (address 0Ch - EPIC I/O address space). Additionally, bit 14 of the Interrupt Mask Register (address 08h - EPIC I/O address space) should be set to one(1) to have the occurrence of a PME assert the PCI Bus interrupt. Writing one(1) to bit 14 in the Interrupt Status Register (address 04h - EPIC I/O address space) will not clear the PME

interrupt in the Interrupt Status Register. It will be necessary to clear the PME status bit in the PMCSR as described above.

### Initial Power-On Reset (POR)

A POR circuit is implemented in the EPIC to detect the initial assertion of power and provide the controller with a reset. The circuit keeps the reset signal in the asserted state until stable power is detected. The POR also clears the PME status and enable bits (bits 15 and 8/PMCSR) and puts the EPIC in the D0 power state (bits 1 and 0/PMCSR set to 00b).

## Special Power Management Mode

A special mode can be enabled to have the EPIC transition to power state D3(Cold1) upon the loss of PCI Bus sourced power to the EPIC.

- a) This special mode can be enabled by setting bit 22 to one(1) in the General Control Register (address 0Ch - EPIC I/O address space). This mode comes up disabled at initial power-up. Therefore, the EPIC maintains the state of this bit through any loss of PCI Bus sourced power and/or the assertion of the PCI Bus reset.
- b) The EPIC is notified of the PCI Bus sourced power loss by way of a board mechanism connecting a "PCI Bus Power Good" indication to the EPIC GPIO1 pin. A high (1) level on this pin indicates the presence of PCI Bus sourced power and a low(0) level on this pin indicates the loss of PCI Bus sourced power.
- c) While in this mode, the transition from PCI Bus sourced power "Not Present" to "Present" will cause the EPIC to transition its power state from D3(Cold1) to D0.
- d) As it would not be possible to configure the EPIC power management capabilities (as described above) after the power state has transitioned to D3 (Cold1), such capabilities should be customized to the particular system environment prior to the loss of PCI Bus sourced power.

## POWER DOWN MODE

The EPIC/100 has a power down feature which allows it to consume less power when not in use. The host may power down the EPIC/100 by writing a 1 to the POWER DOWN bit in the general control register. When the bit is set, the chip's internal system clock is gated off to reduce switching current (the transmit and receive clocks will be shut off internally if the EPIC is in loopback mode when power down is set). While the EPIC/100 is powered down, the host may read and write the configuration registers or the general control register. All other functions are disabled (attempting any other operation will cause unpredictable behavior). The POWER DOWN bit must only be set when the EPIC/100 is in its idle state.

When the nRST pin is asserted, the EPIC will automatically enter power down mode after recalling the contents of the EEPROM. The host may power up the EPIC/100 by writing a 0 to the POWER DOWN bit. If the host wishes to issue a software reset to the EPIC/100, the POWER DOWN bit must be cleared. When the software reset has completed, the POWER DOWN bit will remain cleared and the EPIC/100 will be ready to operate.

The power down bit does not affect the PCI clock inside the EPIC/100. Instead, the EPIC/100 supports the PCI clock run function which allows the host system to slow down or temporarily shut off the PCI clock at its source. The clock run function is implemented according to the PCI Mobile design guide (revision 1.0).

## SOFT RESET

The software driver may reset the LAN83C171 to its initial state by setting the soft reset bit in the general control register. All state machines and pointers to the internal RAMS will be reset. Soft reset can only take place when the LAN83C171 is powered up. (POWER DOWN bit = 0 in general control register). Soft reset does NOT affect the configuration of the LAN83C171. (The configuration registers (excluding EEPROM control) will only be reset and the EEPROM will only be recalled after hard reset).

**Each time the software driver is loaded, it must set soft reset before enabling the LAN83C171 to act as a bus master.** The driver may be loaded after a warm boot, and the LAN83C171 DMA controllers could be left in an unknown state. If the LAN83C171 is enabled as a bus master before a soft reset is issued, the DMA controllers could corrupt host memory with a bus master operation that was started before the warm boot. When the soft reset bit is set, the LAN83C171 takes 15 PCI clocks to re-initialize itself. The device must not be accessed within that time period.

## CONFIGURATION

The LAN83C171 is automatically configured by the host system power-up software before the machine is booted to an operating system. Configuration is performed through the PCI configuration space. The LAN83C171 indicates its requirements and the power-up software allocates the appropriate resources.

The LAN83C171 has the following requirements:

- A) 256 byte I/O space (may be mapped anywhere in 32 bit I/O address space)
- B) 4 Kbyte memory space (may be mapped anywhere in 32 bit host memory space - may be disabled on any host with I/O space).

- C) 64 Kbyte expansion ROM space (may be mapped anywhere in 32 bit host memory space - will be disabled after boot sequence).
- D) Interrupt hardwired to PCI nINTA line.

As resources are relocatable, device drivers must read the configuration registers after boot time to determine where the device is mapped.

## Mapping of Control Functions

The LAN83C171 control registers are mapped into both host I/O and memory space (to accommodate host systems with no I/O space). In machines with I/O space, mapping into memory address space may optionally be disabled by a bit in EEPROM. Any change in the memory map enable bit will not take effect until the system is reset (hard reset). The I/O base address for the control registers will be stored in the I/O Base Address Register (10h in the configuration space). The memory base address for the control registers will be stored in the Memory Base Address Register (14h in the configuration space). The I/O space is 256 bytes long and is always naturally aligned. When mapped into memory, the control registers consume 4 Kbytes even though only the first 256 bytes are used. They are always aligned to a 4 Kbyte boundary. Access to the control registers is enabled by the memory space and I/O space enable bits in the PCI Command Register (04h in the configuration space). Both memory space and I/O space will be disabled after reset. All control registers are dword accessible only.

## Mapping of Flash RAM Functions

The expansion Flash RAM is accessible when the "address decode enable" bit in the ROM base address register (30h in the configuration space) is set and the memory space enable bit is set. The flash RAM code is not executable in place. It must be copied into system RAM and executed from RAM (as per PCI specification).

LAN83C171 control functions will not be accessible through memory address space while the ROM address decode enable bit is set.

When the expansion Flash RAM is read, the LAN83C171 always returns all four bytes in the dword being accessed regardless of the byte enables active. The Flash RAM can also be written by writing to an address in the range defined by the ROM base address register.

The data can be programmed into the flash RAM by following the specifications supplied by the flash manufacturer. The write pulse for each individual byte write will be 200 ns. The EPIC/100 will write all bytes in a dword that have valid byte enables during the data phase. If an access is made to the LAN83C171 while a flash write is physically taking place, a retry will be issued on the PCI bus.

| ROM<br>DECODE<br>ENABLE | MEMORY<br>MAP ENABLE | MEMORY<br>SPACE<br>ENABLE | I/O SPACE<br>ENABLE | EXPANSION<br>ROM | CONTROL<br>REGISTERS |
|-------------------------|----------------------|---------------------------|---------------------|------------------|----------------------|
| 1                       | X                    | 1                         | 1                   | Enabled          | I/O Space            |
| 1                       | X                    | 1                         | 0                   | Enabled          | Disabled             |
| 1                       | X                    | 0                         | 1                   | Disabled         | I/O Space            |
| 1                       | X                    | 0                         | 0                   | Disabled         | Disabled             |
| 0                       | 1                    | 1                         | 1                   | Disabled         | I/O & Memory         |
| 0                       | 1                    | 1                         | 0                   | Disabled         | Mem. Space           |
| 0                       | 1                    | 0                         | 1                   | Disabled         | I/O Space            |
| 0                       | 1                    | 0                         | 0                   | Disabled         | Disabled             |
| 0                       | 0                    | X                         | 1                   | Disabled         | I/O Space            |
| 0                       | 0                    | X                         | 0                   | Disabled         | Disabled             |

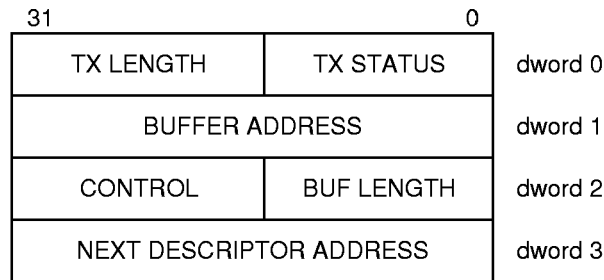
After reset, the ROM will be disabled and the ROM base address will be unknown. The system POST code will map the ROM into host memory space, copy its contents into system RAM and execute the ROM initialization code (if

no ROM signature is found, the system will disable the ROM and proceed with the next PCI device). The ROM initialization code is allowed to perform dynamic re-sizing of the runtime code in order to use as little of the host memory space as possible.

## DMA DESCRIPTOR BITS DESCRIPTION

### TRANSMIT DMA DESCRIPTOR BITS DESCRIPTION

The following diagram shows the format of the transmit descriptor table:



#### DWORD 0 - STATUS/LENGTH Bit Number Description

31 through 16: Transmit Length

15 - OWNER: Descriptor ownership bit - set to 0 when the host owns the descriptor, 1 when the NIC owns the descriptor.

14 and 13 - Reserved.

12 through 8 - COLLISION COUNT: These bits contain the number of collisions detected while attempting to transmit the current packet. Bit 12 also indicates transmit abort for excessive collisions.

7 - DEFERRING: This bit is set when the interframe gap state machine is deferring. If the PHY has asserted the collision line as a result of jabber, this bit will stay set indicating the jabber condition. Otherwise, returns 0.

6 - OUT OF WINDOW COLLISION: This bit is set if a collision is detected more than one slot time after the start of transmission. Transmission is aborted under these conditions.

5 - COLLISION DETECT HEARTBEAT: This bit is set to a '1' during transmission of each packet. It is set to '0' if a collision is detected within 36 bit times of the end of each packet transmission. If no collision is detected within this window, it remains '1'. This bit always returns zero in full duplex mode.

4 - UNDERRUN: This bit is set when the transmit DMA is unable to supply the transmitter enough data to maintain frame transmission. Otherwise, returns 0.

3 - CARRIER SENSE LOST: This bit is set if the carrier is lost during packet transmission. Carrier sense is monitored from its rising edge at the start of the outgoing frame's echo. Transmission is not aborted upon loss of carrier. This bit will always return zero in full duplex mode.

2 - TRANSMITTED WITH COLLISIONS: When set, this bit indicates the frame collided at least once with another frame on the network. It is not set for either out-of-window collisions or excessive collision aborts.

1 - NON-DEFERRED TRANSMISSION: This bit is set if the frame is transmitted successfully without deferring. A deferred transmission can only occur the first time an attempt is made to send a packet. Collisions are not deferred transmissions.

0 - PACKET TRANSMITTED: This bit is set to indicate transmission of a packet without excessive collisions or abort.

**DWORD 1 - DATA BUFFER/START OF FRAGLIST POINTER bit Number Description**

31 through 0: Starting address of data buffer or fragment list in host memory space. Fragment list must be DWORD aligned. Data buffer may be aligned on any byte.

**DWORD 2 - CONTROL/DATA LENGTH Bit Number Description**

31 through 21 Reserved: Must always be set to 0.

20 - LASTDESCR: Indicates that this is the last descriptor for the current transmit frame (Not used when FRAGLIST = 1).

19 - NOCRC: Disable automatic CRC generation for this packet when set.

18 - IAF: When set, interrupt after this frame is transmitted.

17 - LFFORM: Fragment list format - A "1" indicates that the data length field comes before the pointer in the fragment list. "0" indicates that the pointer comes before the data length.

16 - FRAGLIST: Indicates that this descriptor points to a fragment list.

15 through 0 - Length of data buffer (Not used when FRAGLIST = 1).

**DWORD 3 - NEXT DESCRIPTOR POINTER Bit Number Description**

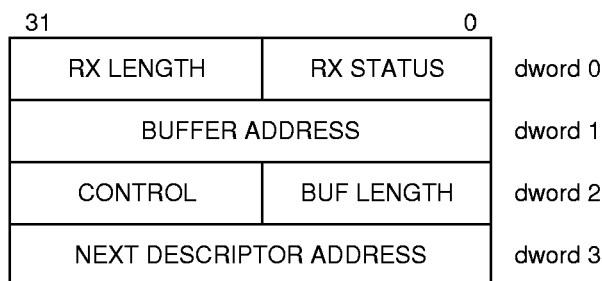
31 through 2 - Starting address of next descriptor in host memory space. Descriptors must be DWORD aligned.

1 - 0: Unused.

## RECEIVE DMA DESCRIPTOR BITS DESCRIPTION

The following diagram shows the format of the

receive descriptor table. Note: The descriptor is not updated when EPIC is put into the monitor mode. This is true only if RXSTAT Register is not read while in the monitor mode.



### DWORD 0 - LENGTH/STATUS Bit Number Description

31 through 16 - RECEIVE FRAME LENGTH: Number of bytes in the received frame.

15 - OWNER: Descriptor ownership bit - set to 0 when the host owns the descriptor, set to 1 when the NIC owns the descriptor.

14 - HEADER COPIED: Set when the receive status is posted after a header copy.

13 - FRAGMENT LIST ERROR: Set when all buffers in the fragment list have been filled before the entire receive frame is copied.

12 - NETWORK STATUS VALID: Set when bits 6 - 0 contain the status from the current frame and bits 31-16 contain the frame length. In the case of a header copy or fragment list error, the receive status from the current frame may or may not be posted. In all other cases this bit will be set.

11 through 7 - Reserved

6 - RECEIVER DISABLED: This bit indicates if the receiver is disabled. This bit is meaningless in monitor mode. (The bit will read back as one

if RXSTAT is read during monitor mode. While in monitor mode, RXSTAT is expected to be read during test only).

5 - BROADCAST ADDRESS RECOGNIZED: This bit is set when a broadcast address has been recognized.

4 - MULTICAST ADDRESS RECOGNIZED: This bit is set when a multicast address which passes the hash filter has been recognized.

3 - MISSED PACKET: This bit is set when a packet with a recognized address and without errors (or with masked errors) is not buffered because the device is in monitor mode. This bit is also set when the packet overflows the receive buffer space and cannot be received. Otherwise, returns 0.

2 - CRC ERROR: This bit is set when a frame's computed CRC does not match the CRC appended to the frame. If the frame is a runt, this bit will be clear. In MII mode, this bit will also be set if receive error was asserted on the MII interface during reception of the frame.

1 - FRAME ALIGNMENT ERROR: This bit is set if a CRC error has occurred and the frame is not byte aligned.

0 - PACKET RECEIVED INTACT: This bit is set when a packet is received into the buffer space without error.

**DWORD 1 - DATA BUFFER/START OF FRAGLIST POINTER Bit Number Description**

31 through 0 - Starting address of data buffer or fragment list in host memory space. Fragment list must be DWORD aligned. Data buffer may be aligned on any byte.

**DWORD 2 - CONTROL/DATA LENGTH OR FRAME OFFSET Bit Number Description**

31 through 19 - Reserved: Must always be set to 0.

18 - HEADER: Indicates that this descriptor is for a header copy.

17 - LFFORM: Fragment list format - a 1 indicates that the data length field comes before the pointer in the fragment list. A 0 indicates that the pointer comes before the data length.

16 - FRAGLIST: Indicates that this descriptor points to a fragment list.

15 through 0 - Length of data buffer (when FRAGLIST = 0) or Offset into frame where copy begins (when FRAGLIST = 1).

**DWORD 3 - NEXT DESCRIPTOR POINTER Bit Number Description**

31 through 2 - Starting address of next descriptor in host memory space. Descriptors must be DWORD aligned.

1 - 0: Not used.

## CONTROL REGISTER MAP/REGISTERS DECODE

The following table shows the address mapping of the LAN83C171 control registers. All registers are dword accessible only.

|    |                       |    |                     |    |                      |    |                      |
|----|-----------------------|----|---------------------|----|----------------------|----|----------------------|
| 00 | COMMAND               | 40 | LAN0 <sup>1</sup>   | 80 | PRFDAR <sup>1</sup>  | C0 | PTFDAR <sup>1</sup>  |
| 04 | INTSTAT               | 44 | LAN1 <sup>1</sup>   | 84 | PRCDAR <sup>1</sup>  | C4 | PTCDAR <sup>1</sup>  |
| 08 | INTMASK               | 48 | LAN2 <sup>1</sup>   | 88 | PRHDAR <sup>4</sup>  | C8 | PTH DAR <sup>4</sup> |
| 0C | GENCTL                | 4C | ID/CHK <sup>1</sup> | 8C | PRFLAR <sup>4</sup>  | CC | PTFLAR <sup>4</sup>  |
| 10 | NVCTL                 | 50 | MC0 <sup>1</sup>    | 90 | PRDLGTH <sup>4</sup> | D0 | PTDLGTH <sup>4</sup> |
| 14 | EECTL                 | 54 | MC1 <sup>1</sup>    | 94 | PRFCNT <sup>4</sup>  | D4 | PTFCNT <sup>4</sup>  |
| 18 | PBLCNT                | 58 | MC2 <sup>1</sup>    | 98 | PRLCAR <sup>4</sup>  | D8 | PTLCAR <sup>4</sup>  |
| 1C | Reserved <sup>4</sup> | 5C | MC3 <sup>1</sup>    | 9C | PRLPAR <sup>4</sup>  | DC | ETXTHR <sup>1</sup>  |
| 20 | CRCNT                 | 60 | RXCON <sup>1</sup>  | A0 | PREFAR <sup>4</sup>  | E0 | PTETXC <sup>4</sup>  |
| 24 | ALICNT                | 64 | RXSTAT <sup>4</sup> | A4 | PRSTAT <sup>3</sup>  | E4 | PTSTAT <sup>4</sup>  |
| 28 | MPCNT                 | 68 | RXCNT <sup>4</sup>  | A8 | PRBUF <sup>4</sup>   | E8 | PTBUF <sup>4</sup>   |
| 2C | RXFIFO <sup>4</sup>   | 6C | RXTEST <sup>4</sup> | AC | RDNCAR <sup>4</sup>  | EC | PTFDAR2 <sup>4</sup> |
| 30 | MMCTL                 | 70 | TXCON <sup>1</sup>  | B0 | PRCPTHR <sup>1</sup> | F0 | FEVTR                |
| 34 | MMDATA                | 74 | TXSTAT <sup>2</sup> | B4 | ROMDATA              | F4 | FEVTRMSKR            |
| 38 | MIICFG <sup>1</sup>   | 78 | TDPAR <sup>4</sup>  | B8 |                      | F8 | FPRSTSTR             |
| 3C | IPG <sup>1</sup>      | 7C | TXTEST <sup>4</sup> | BC | PREEMPR <sup>1</sup> | FC | FFRCEVTR             |

Notes:

<sup>1</sup>Used only during initialization (illegal to access when not idle).

<sup>2</sup>Legal to access during transmit underrun only.

<sup>3</sup>Legal to access only when frame is discarded after header copy and INTSTAT.RSV is 1.

<sup>4</sup>Legal to access only in test mode.

## CONFIGURATION REGISTERS MAP

The following table shows the address mapping for the LAN83C171 configuration registers.

Registers are byte, word or dword accessible (reads always return all four bytes).

|               |                                   |                     |                       |                      |   |
|---------------|-----------------------------------|---------------------|-----------------------|----------------------|---|
| 31            |                                   |                     |                       |                      | 0 |
| 00            | Device ID                         |                     |                       | Vendor ID            |   |
| 04            | Status                            |                     |                       | Command              |   |
| 08            | Class Code                        |                     |                       | Rev ID               |   |
| 0C            | Unused                            | HDR Type            | LAT Timer             | Unused               |   |
| 10            | I/O Base Address                  |                     |                       |                      |   |
| 14            | Memory Base Address               |                     |                       |                      |   |
| 18            | Unused                            |                     |                       |                      |   |
| 1C            | Unused                            |                     |                       |                      |   |
| 20            | Unused                            |                     |                       |                      |   |
| 24            | Unused                            |                     |                       |                      |   |
| 28            | CIS Pointer                       |                     |                       |                      |   |
| 2C            | Subsystem ID                      |                     | Subsystem Vendor ID   |                      |   |
| 30            | Expansion ROM Base Address        |                     |                       |                      |   |
| 34            | Reserved                          |                     |                       |                      |   |
| 38            | Reserved                          |                     |                       |                      |   |
| 3C            | Max Lat                           | Min Gnt             | Int. Pin              | Int. Line            |   |
| 40<br>:<br>D8 | Unused                            |                     |                       |                      |   |
| DC            | Power Management(PM) Capabilities |                     | Next Capabilities Ptr | PMRB Capabilities ID |   |
| E0            | Unused                            | PM Control / Status |                       |                      |   |
| E4<br>:<br>FF | Unused                            |                     |                       |                      |   |

Note: All unused and reserved registers return zeroes, when read. Writes to unused and reserved registers are ignored.

## CONTROL REGISTERS BITS DESCRIPTION

### 00 - COMMAND Register

Reset Value: 00000000

31 through 8 - Reserved: These bits will return unknown values and should never be written to. 7 - TXUGO: This bit is set to restart transmission after a transmit underrun error. Setting(1) this bit automatically clears the transmit underrun interrupt. Writing a zero to this bit has no effect. This bit always returns 0 when read.

6 - STOP\_RDMA: This bit is used to halt the receive DMA. Writing a 1 to this bit clears RXQUEUED. Writing a 0 to this bit has no effect. This bit always returns 0 when read.

5 - STOP\_TDMA: This bit is used to halt the transmit DMA. Writing a 1 to this bit clears TXQUEUED. Writing a 0 to this bit has no effect. This bit always returns 0 when read.

4 - NEXTFRAME: This bit is set by the host to indicate that it does not need any more copies of the current receive frame. The bit will be cleared by the LAN83C171 the next time it reads a descriptor. Writing a 0 to this bit has no effect (in register test mode writing 0 clears the bit.)

3 - RXQUEUED: This bit is set to queue a receive descriptor. It will be cleared by the LAN83C171 when it reads a descriptor that is still owned by the host. Setting this bit automatically clears the receive queue empty interrupt. Writing a 0 to this bit has no effect (in register test mode writing 0 clears the bit.) The host may clear this bit by writing a 1 to RDMA\_STOP.

2 - TXQUEUED: This bit is set to queue a transmit descriptor. It will be cleared by the LAN83C171 when it reads a descriptor that is still owned by the host. Setting this bit automatically clears the transmit queue empty interrupt. Writing a 0 to this bit has no effect (in

register test mode writing 0 clears the bit). The host may clear this bit by writing a 1 to TDMA\_STOP.

1 - START\_RX: Writing a 1 to this bit will bring the LAN83C171 receiver on line. When this bit is cleared, the receiver will stay online until the stop bit is set.

0 - STOP\_RX: Writing a 1 to this bit will take the LAN83C171 receiver off line. When this bit is cleared, the receiver will stay offline until the start bit is set.

### 04 - INTERRUPT STATUS Register

Reset Value: 010011000000000000000000

31 through 28: Unused.

27 - PTA: PCI Target abort - set when EPIC/100 cannot complete a bus master transaction because target aborts the transaction.

26 - PMA: PCI Master abort - set when EPIC/100 cannot complete a bus master transaction because no target is found.

25 - APE: PCI address parity error - set when an address parity error occurs on the PCI bus while EPIC/100 is not bus master. This interrupt will only be set when the Parity Error Response bit in the PCI configuration space is set.

24 - DPE: PCI data parity error - set when a data parity error occurs on the PCI bus while EPIC/100 is bus master. This interrupt will only be set when the Parity Error Response bit in the PCI configuration space is set.

23 - RSV: Receive status valid (read only - does not generate an interrupt) - indicates that the PRSTAT register contains valid status for the frame currently being processed.

22 - RCTS: Receive copy threshold status (read only - does not generate an interrupt) - indicates that the copy in progress that has passed the early receive copy threshold. This bit returns zero when there is no receive copy currently in progress or when the current copy has not passed the threshold.

21 - RBE: Receive buffers empty (read only - does not generate an interrupt) - indicates that there is no data ready for copy in the receive buffer.

20 - TCIP: Transmit copy in progress (read only - does not generate an interrupt) - indicates that a transmit DMA copy is partially completed. The bit is set each time the receive DMA begins to copy a frame. The bit is reset after the copy completes and the status is posted.

19 - RCIP: Receive copy in progress (read only - does not generate an interrupt) - indicates that receive DMA copy is partially completed. The bit is set each time the receive DMA begins to copy a frame. The bit is reset after the copy completes and the status is posted.

18 - TXIDLE: Transmit idle (read only - does not generate an interrupt) - indicates that the NIC transmitter and PCI transmit DMA have returned their reset states.

17 - RXIDLE: Receive idle (read only - does not generate an interrupt) - indicates that the NIC receiver and PCI receive DMA have returned their reset states.

16 - INT\_ACTV: Interrupt active (read only - does not generate an interrupt) - indicates that an interrupt which is not masked is currently set. This allows the host to read the interrupt status through a register, even when interrupts are disabled.

15 - GP2\_INT: This interrupt becomes active when the pin GPIO(2) goes low. It is typically used by the PHY to indicate an event.

14 - PME: Power management event (read only) - indicates that a "Magic Packet" and/or a link-down event has occurred during one of the low power states. This bit may only be cleared by writing a 1 to bit 15 of the PMCSR.

13 - unused.

12 - FATAL\_INT: This signal becomes true if any fatal error occurs. These are DPE, APE, PMA, and PTE. Note that these are also reflected in the interrupt status word (27:24).

11 - RCT: Receive copy threshold crossed - set when the receive copy in progress crosses the PCI receive copy threshold.

10 - PREI: This preemptive interrupt event indicates that a packet is being received, with the probability that by the time the host responds to the interrupt, the packet will have been completely received, reducing latency.

9 - CNT: Counter overflow - indicates that one of the error counters is nearing its maximum count.

8 - TXU: Transmit underrun - set when an early transmit underrun occurs. This interrupt is cleared automatically when the TXUGO bit in the command register is set. Clearing this interrupt manually (by writing to this register) does not effect the TXUGO bit.

7 - TQE: Transmit queue empty - set when NIC reads a transmit descriptor that is still owned by the host. This interrupt is cleared automatically when the TXQUEUED bit in the command register is set. Clearing this interrupt manually (by writing to this register) does not effect the TXQUEUED bit.

6 - TCC: Transmit chain complete - set when the complete transmit chain has been processed.

5 - TXC: Transmit complete - set when a packet has been successfully transmitted or aborted and the IAF bit is set for that frame.

4 - RXE: Receive error - set when a CRC error occurs and Monitor mode is off.

3 - OVW: Receive buffer overflow warning - set when a frame is received and local receive buffer space is full.

2 - RQE: Receive queue empty - set when NIC reads a receive descriptor that is still owned by the host. This interrupt is cleared automatically when the RXQUEUED bit in the command register is set. Clearing this interrupt manually (by writing to this register) does not effect the RXQUEUED bit.

1 - HCC: Header copy complete - set when receive frame header has been copied into host memory.

0 - RCC: Receive copy complete - set when receive frame has been copied into host memory.

#### **08 - INTERRUPT MASK Register**

Reset Value: 0000000000000000

This register is used to enable certain interrupt sources selectively. Bits that are 1 allow the corresponding interrupt to cause an interrupt request. Bits that are 0 block their interrupt sources.

31 through 16: Unused.

15 through 0: Interrupt enables.

#### **0C - GENERAL CONTROL**

Reset Value: 000000000000001000000000

(Note: The POWER DOWN bit will automatically be set following the EEPROM recall when the nRST pin is asserted, it will also be set when the EPIC is placed in one of the low power states. The POWER DOWN bit will remain clear after a soft reset.)

31 through 23 - Unused.

22 - D3 ON POWER LOSS ENABLE: This bit enables the EPIC to enter the D3 power state upon the loss of PCI Bus sourced power.

21 and 20 - POWER STATE RESET SELECT: These bits control the effect of the PCI Bus reset on the EPIC power state as follows:

- 1) "00" - The assertion of PCI Bus reset will immediately transition the EPIC power state to D0. At initial power-up, this is the state of these two bits.
- 2) "01" - The assertion of the PCI Bus reset will not have any effect on the EPIC power state. However, the transition from the PCI Bus reset being asserted to deasserted will change the EPIC power state to D0.
- 3) "10" - Transitions of the PCI Bus reset will not have any effect on the EPIC power state.
- 4) "11" - Same as "10".

19 - PME INTA ENABLE: This bit enables the occurrence of an enabled PME during one of the low power states to set bit 14 of the Interrupt Status register.

18 - MAGIC PACKET ENABLE: This bit enables the detection of a "Magic Packet" to set the PME status bit in the PMCSR.

17 - RESET DISABLE: This bit when set disables the effects of the PCI Bus reset on the EPIC.

16 - SCLK ENABLE: This bit, when set, restores the EPIC internal system clock when the POWER DOWN bit is set.

15 - Unused.

14 - RESET PHY: This bit is OR'ed with the PCI nRST input to generate the nPHYRST output for the physical layer device.

13 and 12 - SOFT[1:0]: These two read/write bits are provided for use by the software driver. They do not affect hardware operation.

11 and 10 - MEMORY READ CONTROL: These bits control which PCI command the transmit DMA will use when bursting data over the PCI Bus. When bit 11 is set, the transmit DMA will use the PCI "memory read line" command. When bit 10 is set, the transmit DMA will use the PCI "memory read multiple" command. When neither bit is set the transmit DMA will use the PCI "memory read" command. Use of "memory read multiple" or "memory read line" may enhance performance on some machines.

9 and 8 - RECEIVE FIFO THRESHOLD: Controls the level at which the PCI burst state machine begins to empty the receive FIFO. Default is  $\frac{1}{2}$  full. D9 = THR\_SEL[1], D8 = THR\_SEL[0].

| [9] | [8] | THRESHOLD LEVEL     |
|-----|-----|---------------------|
| 0   | 0   | 1/4 Full (32 Bytes) |
| 0   | 1   | 1/2 Full (64 Bytes) |
| 1   | 0   | 3/4 Full (96 Bytes) |
| 1   | 1   | Full (128 Bytes)    |

7 - TRANSMIT DMA PRIORITY: When this bit is set, the transmit DMA may preempt the receive DMA for access to the PCI Bus. Preemption occurs when the PCI latency timer expires.

6 - RECEIVE DMA PRIORITY: When this bit is set, the receive DMA may preempt the transmit DMA for access to the PCI Bus. Preemption occurs when the PCI latency timer expires.

5 - BIG ENDIAN: This bit controls the order of the bytes on the data bus when the EPIC is used in a big endian machine. When this bit is set to a 1, the EPIC performs byte swapping on the descriptor and fragment list entries to compensate for byte swapping by the PCI bridge.

4 - ONECOPY: When this bit is set to a 1, the EPIC will give the host only one copy of each receive frame. This bit causes NEXTFRAME to be set automatically at the end of each frame. This bit should not be modified when the receive DMA is not idle.

3 - POWER DOWN: Setting this bit puts the EPIC into a low power sleep mode. When this bit is cleared (I/O writes to this register are still enabled in sleep mode) the EPIC will resume in the state it was in prior to the power down. This bit may only be set when the chip is idle.

2 - SOFTWARE INTERRUPT: When this bit is set to a 1, the EPIC interrupt pin nINTA will become active (driven low).

1 - INTERRUPT ENABLE: Setting this bit enables the EPIC interrupt line. When one of the interrupt status bits and its corresponding mask bit are both set, the EPIC will drive the nINTA pin low. Clearing this bit masks all interrupts (except software interrupt).

0 - SOFT RESET: Setting this bit to a 1 resets the EPIC to its initialization state. All state machines and pointers to the internal RAMs will be reset. The configuration registers and the non-volatile control register will not be reset and EEPROM recall will not take place after a soft reset. This register will return to its reset value after the operation is complete, regardless of the data written. The POWER DOWN bit must also be cleared for the soft reset to take effect.

#### **10 - NON-VOLATILE CONTROL Register**

Reset Value (Power Up): 000000000000

This register is loaded from the EEPROM after reset.

31 through 11: Unused.

10 through 7 - IPG\_DLY: This register holds the inter-packet gap timer. It is used to increase the amount of delay between packets as they are being transmitted to decrease the chance of collisions. The delays can vary between 0 and 19.2  $\mu$ s in 1.28  $\mu$ s intervals.

6 - CB\_MODE: This bit designates whether the Epic is in a PCI or a CardBus environment. A one (1) is CardBus mode while a zero (0) is PCI mode. It determines whether an interrupt comes straight out from the interrupt status lines, or if the CardBus functional event register is used.

5 - GENERAL PURPOSE I/O[2]: This bit controls the value of the GPIO[2] pin when used as an output. When read, this bit always returns the external value on GPIO[2].

4 - GENERAL PURPOSE I/O[1]: This bit controls the value of the GPIO[1] pin when used as an output. When read, this bit always returns the external value on GPIO[1].

3 - GENERAL PURPOSE OUTPUT ENABLE[2]: When set, GPIO[2] is driven by the LAN83C171. When cleared, GPIO[2] is tri-stated and may be used as an input. When used as an input, this pin is reflected in INTSTAT(15), and is normally used as an interrupt indication from the physical layer.

2 - GENERAL PURPOSE OUTPUT ENABLE[1]: When set, GPIO[1] is driven by the LAN83C171. When cleared, GPIO[1] is tri-stated and may be used as an input.

1 - CLOCK RUN SUPPORTED: This bit enables the LAN83C171 to perform the PCI clock run function. When set, the clock run function is enabled. When cleared, the nCLKRUN output is tri-stated. This bit is only writable in register test mode. During normal operation, it should only be changed by re-programming the EEPROM and resetting the system (hard reset.)

0 - ENABLE MEMORY MAP: This bit controls whether or not the LAN83C171 control registers are visible in memory space. When set, the LAN83C171 control registers will be mapped into I/O space and memory space (for host systems that do not have I/O space). When cleared, the control registers will only be

mapped into I/O space. This bit controls how the host system maps the control registers at power up by changing the appearance of the memory base address register in PCI configuration space. This bit is only writable in register test mode. In normal operation, it should only be changed by re-programming the EEPROM and resetting the system (hard reset.) Default is disabled when EEPROM recall is bypassed.

#### **14 - EEPROM CONTROL Register**

Reset Value: XXX0000

6 - EEPROM SIZE: This read only bit indicates the size of the external serial EEPROM (1 = 16x16 or 64x16, 0 = 128x16 or 256x16). The size is selected by an external "jumper" at power-on reset.

5 - EERDY: This read only bit indicates when the EEPROM input data is valid and/or when any of the EEPROM outputs may be changed (1 = ready, 0 = not ready.)

4 - EEDO: DATA output from EEPROM - Used to read back data from serial EEPROM. This bit is wired directly to the MD[31] input.

3 - EEDI: Data input to EEPROM - Used to supply address and data to serial EEPROM. This bit is muxed onto MA[13] when EEPROM ENABLE is set.

2 - EESK: EEPROM clock - Used to supply the clock to the serial EEPROM. The value of this bit is muxed onto MA[14] when EEPROM ENABLE is set.

1 - EECS: EEPROM chip select - This bit is wired directly to the EECS output pin on the LAN83C171.

0 - EEPROM ENABLE: When this bit is set, EESK and EEDI are multiplexed onto the MA pins.

#### **18 - PBLCNT Register**

Reset Value: 000000

This programmable burst length counter sets a maximum number of dwords that can be transferred during any DMA read or write by the EPIC/100.

5 through 0 - PBLCNT: The value in this register reflects the maximum number of dwords allowed to be transferred in a read or write burst.

#### **18 - Reserved**

Do not write to this address.

#### **20 - CRC ERROR COUNTER Register**

Reset Value: 00000000

31 through 8: Unused.

7 through 0: Reports the number of CRC errors since the last time this register was read. The count will stay at 255 when reached the max. When the count reaches 192, the counter overflow interrupt will be set. The count is cleared when read.

#### **24 - FRAME ALIGNMENT ERROR COUNTER Register**

Reset Value: 00000000

31 through 8: Unused.

7 through 0: Reports the number of frame alignment errors since the last time this register was read. The count will stay at 255 when reached the max. When the count reaches 192, the counter overflow interrupt will be set. The count is cleared when read.

#### **28 - MISSED PACKET COUNTER Register**

Reset Value: 00000000

31 through 8: Unused.

7 through 0: Reports the number of missed packet errors since the last time this register was read. The count will stay at 255 when reached the max. When the count reaches 192, the counter overflow interrupt will be set. The count is cleared when read.

### **30 - MII MANAGEMENT INTERFACE CONTROL Register**

Reset Value: 0000000000000000

This register provides management interface control for functions such as read, write, and synchronize. It also contains the PHY address field and the PHY register address field - to be sent in the command word to the PHY. A management operation is executed by writing the corresponding operation bit to this register. When the operation is complete, the bit is cleared automatically.

31 through 14: Unused.

13 through 9 - PHY ADDRESS FIELD: This 5 bit field is sent in the management frame to the PHY. D13 is the MSB.

8 through 4 - PHY REGISTER ADDRESS FIELD: This 5-bit field is sent in the management frame to the PHY. D8 is the MSB.

3 - RESPONDER: This bit returns a 1 during a read operation if a PHY responded with a zero level on the MDIO line during the first SMCLK cycle following the idle bit time when both, the management entity and the PHY, do not drive the MDIO. This bit can be used to determine if a PHY responded to the read operation. This bit is self clearing following a register read. This bit is read only.

2 - Unused.

1 - WRITE: This bit is set to 1 to initiate a write operation on the management interface. When set, a properly formatted management frame is sent to the PHY. The data field of the

management frame is filled with the contents of the Management Interface Data register. The bit is self clearing after completion of the operation.

0 - READ: This bit is set to 1 to initiate a read operation on the management interface. When set, a properly formatted management frame will be sent on the MDIO line with corresponding cycles on MDC. Data returned by the PHY is shifted into the Management Interface Data register. The bit is self clearing after completion of the operation.

### **34 - MII MANAGEMENT INTERFACE DATA Register**

Reset Value: 0000000000000000

This 16 bit register is used by the MII management unit for all data transfers between the management and PHY(s).

31 through 16: Unused.

15 through 0 - FRAME DATA: A 16 bit value written to this register will be used in the data field of a management interface write operation. For read operations, this 16 bit value will store the data transferred from the PHY.

### **38 - MII CONFIGURATION Register**

Reset Value: 0001XX00

This register provides MII configuration functions.

31 through 16: Unused.

15 - SET NO TX CLOCK: When bit 14 of this register is set to one (1), bit 15 will specify the clock source. Setting this bit to one (1) will force the TX clock based internal registers to use the inverted SYSCLK as a clock reference instead of the TX\_CLK input. SYSCLK being gated-off in the specified powerdown modes, will help conserve current for the TX clock dependent registers also.

14 - SELECT TX CLOCK F/F: See bit 15 description.

13 through 8: Unused.

7 - ALTERNATE DIRECTION: When set, the alternate data value is input from the MII management data pin, if serial management interface is disabled.

6 - ALTERNATE DATA: Reading this bit returns the value at the MII management data pin. A value written to this bit will be driven onto the MII management data pin when the serial management interface is disabled and the alternate direction bit is set to output.

5 - ALTERNATE CLOCK SOURCE: This register bit is muxed to the MII management clock pin when the serial management interface is disabled. When set, the management interface clock is set.

4 - ENABLE SERIAL MANAGEMENT INTERFACE: This bit selects between the serial management interface and a general purpose interface muxed with the management interface clock and data pins. When set, the serial management interface is selected. The default for this bit is set (1).

3 - PHY PRESENT: This bit is read only. It is set to one when the MDIO line is at a logic one value indicating the presence of a PHY device.

2 - 694 LINK STATUS: This bit is read only and returns the value of the 694LNK pin on the LAN83C171.

1 - ENABLE 694: When set, the EN694 pin of the LAN83C171 is driven to a logic one. When clear, the EN694 pin is driven low.

0 - SERIAL MODE ENABLE: When set, the MII interface functions serially as a 7-wire interface. This mode should be enabled when the LAN83C171 is connected to a 10Mb/s serial

PHY device. When clear, the MII interface operates as defined by the IEEE 802.3u Reconciliation Sublayer and Media Independent Interface(MII) Draft Standard.

### 3C - INTERPACKET GAP Register

Reset Value: 011110001100000

This register is used to program the interpacket gap protocol timer. It contains two values. The first 8 bit value is used to set the total interpacket gap time used by the transmit state machine for deferral. The second 7-bit value sets the first interframe spacing value used in the deference process.

31 through 15: Unused.

14 through 8 - INTERFRAME SPACING -PART ONE: This 7 bit value sets the first part of the interframe spacing delay time. The default is 60 bit times.

7 through 0 - INTERPACKET GAP TIME: This 8 bit value sets the interpacket gap delay time. The default is 96 bit times.

### 40 through 48 - LAN ADDRESS Registers

Reset Value (Power Up): Unknown

These registers hold the 48-bit LAN address for the adapter. They are recalled from the EEPROM after (hard)reset.

31 through 16: Unused.

15 through 0 - LAN ADDRESS: The Destination address described as:

[N1][N0][N3][N2][N5][N4][N7][N6][N9][N8][N11][N10]

Where each N is one nibble it will be mapped to the LAN address registers as follows:

LAN0 [15-12] = N3

LAN0 [11-8] = N2

LAN0 [7-4] = N1

LAN0 [3-0] = N0  
 LAN1 [15-12] = N7  
 LAN1 [11-8] = N6  
 LAN1 [7-4] = N5  
 LAN1 [3-0] = N4  
 LAN2 [15-12] = N11  
 LAN2 [11-8] = N10  
 LAN2 [7-4] = N9  
 LAN2 [3-0] = N8

#### 4C - BOARD ID/CHECKSUM Register

Power Up Reset Value: XXXXXXXXXXXXXXXX

This register holds the board ID and the checksum for the adapter. It is recalled from the EEPROM after reset.

31 through 16: Unused.

15 through 8 - BOARD ID: Used as the 8 bit LAN adapter ID field.

7 through 0 - CHECKSUM: Used as the checksum for the LAN address and board ID. The sum of the 6 LAN address bytes, the board ID, and the checksum should be FF.

#### 50 through 5C - MULTICAST ADDRESS HASH TABLE Registers

Reset Value: Unknown

These 4 registers hold the node's multicast filter table.

31 through 16 (MC0): Unused.

15 through 0 - HASH TABLE: The bits in the hash table are decoded in the following order:

MC0 = 15-0  
 MC1 = 31-16  
 MC2 = 47-32  
 MC3 = 63-48

#### 60 - RECEIVE CONTROL Register

Reset Value: XX00000000

31 through 10: Unused.

9 and 8 - EXTERNAL BUFFER SIZE SELECT: When D9:8 = <00>, external buffer access is disabled and all packets are buffered internally. D9:8 = <01> -> 16Kbyte, D9:8 = <10> -> 32Kbyte, and D9:8 = <11> -> 128Kbyte. These bits are jumper set on reset.

7 - EARLY RECEIVE ENABLE: When set, the receiver operates in early receive mode. When early receive is enabled, save errored packets must be set. The runt size (slot time) must be programmed to a value greater than or equal to 224 bit times.

6 - MONITOR MODE: Disables the buffering of receive packets. Receive status and counters continue to function. Receive error interrupt is not posted in monitor mode.

5 - PROMISCUOUS MODE: When set, address filtering is bypassed and all frames with individual addresses are received.

4 - RECEIVE INVERSE INDIVIDUAL ADDRESS FRAMES: When set, individually addressed frames are received that do NOT match the programmed LAN address register.

3 - RECEIVE MULTICAST FRAMES: When set, multicast address filtering is enabled. Frames with multicast addressing and that pass the multicast hash filter, will be received.

2 - RECEIVE BROADCAST FRAMES: When set, broadcast frames are received.

1 - RECEIVE RUNT FRAMES: When set, frames less than one slot time (in length) are received.

0 - SAVE ERRORED PACKETS: When set, frames with CRC and alignment errors are saved in the receive buffers.

## 64 - RECEIVE STATUS

Reset Value: 0000000

The receive status register reports the status of the most-recently received packet. It reports receive errors and address recognition type. All bits are cleared at the start of reception except for receiver disabled. The contents of the lower order bits in this register ([6:0]) make up the lower order bits of the receive packet stamp in the receive buffer.

31 through 7 - Unused.

6 - RECEIVER DISABLED: This bit is set when the receiver is in monitor mode.

5 - BROADCAST ADDRESS RECOGNIZED: This bit is set when a broadcast address has been recognized.

4 - MULTICAST ADDRESS RECOGNIZED: This bit is set when a multicast address which passes the hash filter has been recognized.

3 - MISSED PACKET: This bit is set when a packet with a recognized address and without errors (or with masked errors) is not buffered because the device is in monitor mode. This bit is also set when the packet overflows the receive buffer space and cannot be received. Always returns 0.

2 - CRC ERROR: This bit is set when a frame's computed CRC does not match the CRC appended to the frame. If the frame is a runt, this bit will be clear. In MII mode, this bit will also be set if receive error was asserted on the MII interface during reception of the frame.

1 - FRAME ALIGNMENT ERROR: This bit is set if a CRC error has occurred and the frame is not byte aligned.

0 - PACKET RECEIVED INTACT: This bit is set when a packet is received into the buffer space without error.

## 68 - RECEIVE BYTE COUNT

Reset Value: 0000000000000000

This 16 bit register contains the receive byte count for the most recently received frame. It is cleared by the receive unit at the start of reception of each frame.

5-0 - RECEIVE BYTE COUNT: D15 is the MSB and D0 is the LSB.

## 6C - RECEIVE TEST

Reset Value: 00000xx00000000

31 through 15 - Unused.

14 through 10 - RECEIVE FIFO LEVEL: This 5 bit value returns the receive fifo level.

9 and 8 - Unused.

7 - RUNT STATUS: Returns 0 when the current reception is not a runt or receive runt frames is set. Returns zero when the current receive byte count is less than the runt size. This bit is read only.

6 through 0 - Reserved: Do not write '1' to these bits.

## 70 - TRANSMIT CONTROL Register

Reset Value: 01111000

31 through 8: Unused.

7 through 3 - SLOT TIME: Selects the number of bit times to use for the slot time. The value programmed plus one is multiplied by 32 to generate the slot time. This value is used for both the backoff timer and for runt checking. Default is 0Fh which gives a slot time of 512 bit times.

2 and 1 - LOOPBACK MODE SELECT:

| <u>D2</u> | <u>D1</u> | <u>Mode</u>                                                                                                        |
|-----------|-----------|--------------------------------------------------------------------------------------------------------------------|
| 0         | 0         | Normal operation.                                                                                                  |
| 0         | 1         | Internal loopback. Packets transmitted are internally looped back to the receiver without transmission to the MII. |
| 1         | 0         | External loopback. Turns on the external loopback mode to signal the PHY to loop back transmit packets.            |
| 1         | 1         | Full Duplex mode. Decouples transmit and receive blocks to allow full duplex operation without collisions.         |

0 - EARLY TRANSMIT ENABLE: When set, the transmitter operates in early transmit mode.

#### 74 - TRANSMIT STATUS Register

Reset Value: 0000000000000

The transmit status register reports events that occur on the media at the end of packet transmission. All bits are cleared prior to transmission of a packet and are set as needed. This register may be read when a transmit underrun occurs (before the TXUGO bit is set), otherwise it should only be accessed for test purposes.

31 through 13: Unused.

12 through 8 - COLLISION COUNT: These bits contain the number of collisions detected while attempting to transmit the current packet. Bit 12 also indicates transmit abort for excessive collisions.

7 - DEFERRING: This bit is set when the interframe gap state machine is deferring. If the PHY has asserted the collision line as a result of jabber, this bit will stay set indicating the jabber condition. Always returns 0.

6 - OUT OF WINDOW COLLISION: This bit is set if a collision is detected more than one slot

time after the start of transmission. Transmission is aborted under these conditions.

5 - COLLISION DETECT HEARTBEAT: This bit is set to a '1' during transmission of each packet. It is set to '0' if a collision is detected within 36 bit times of the end of each packet transmission. If no collision is detected within this window, it remains '1'. This bit always returns zero in full duplex mode.

4 - UNDERRUN: This bit is set when the transmit DMA is unable to supply the transmitter enough data to maintain frame transmission.

3 - CARRIER SENSE LOST: This bit is set if the carrier is lost during packet transmission. Carrier sense is monitored from its rising edge at the start of the outgoing frames echo. Transmission is not aborted upon loss of carrier. This bit will always return zero in full duplex mode.

2 - TRANSMITTED WITH COLLISIONS: When set, this bit indicates the frame collided at least once with another frame on the network. It is not set for out-of-window collisions or excessive collision aborts.

1 - NON-DEFERRED TRANSMISSION: This bit is set if the frame was transmitted successfully without deferring. A deferred transmission can only occur the first time an attempt is made to send a packet. Collisions are not deferred transmissions.

0 - PACKET TRANSMITTED: This bit is set to indicate transmission of a packet without excessive collisions or abort.

#### 78 - TRANSMIT PACKET ADDRESS

Reset Value: 000000000

This register contains the transmit MTU's pointer to the starting address of the current frame in the local transmit ram. The register

contains the dword address and is write only. Reads to this register return unknown data.

31 through 9 - Unused.

8 through 0 - Address.

#### **7C - TRANSMIT TEST**

31 through 12 - Unused.

11 through 8 - Reserved: Do not write '1' to these bits.

7 - Force collision.

6 through 0 - Reserved: Do not write '1' to these bits.

#### **80 - PCI RECEIVE FIRST DESCRIPTOR ADDRESS**

Reset Value:

xxxxxxxxxxxxxxxxxxxxxxxxxxxx00

This register contains the byte address of the first descriptor for the current receive packet. It is the location in host memory where the receive status will be posted when the receive copy is complete. The two LSBs are fixed at zero so that the address will always be DWORD aligned. This register is automatically written with the same data as the PRCDAR register whenever a write to that register occurs.

31 through 2 - PCI ADDRESS.

1 and 0 - Not writable - always return zeroes.

#### **84 - PCI RECEIVE CURRENT DESCRIPTOR ADDRESS**

Reset Value:

xxxxxxxxxxxxxxxxxxxxxxxxxxxx00

This register contains the byte address (in host memory) of the next descriptor that the receive DMA will read. The two LSBs are fixed at zero so that the address will always be DWORD aligned. Data written into this register will automatically be stored in the PRFDAR register at the same time.

31 through 2 - PCI ADDRESS.

1 and 0 - Not writable - always return zeroes.

#### **88 - PCI RECEIVE HOST DATA ADDRESS**

Reset Value:

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

This register contains the address where the receive packet data is to be written in host memory. The upper 30 bits are driven onto the PCI Bus as the DWORD address, and are incremented each time a DWORD is written to host memory. The two LSBs always contain the starting byte address of the data buffer, and are used by the receive DMA to control byte alignment.

31 through 2 - PCI ADDRESS.

1 and 0 - STARTING BYTE ADDRESS.

#### **8C - PCI RECEIVE FRAGMENT LIST ADDRESS**

Reset Value:

xxxxxxxxxxxxxxxxxxxxxxxxxxxx00

This register contains the current fragment list address. It is the location in host memory of the next fragment list entry that the receive DMA will read. The two LSBs are fixed at zero so that the address will always be DWORD aligned.

31 through 2 - ADDRESS.

1 and 0 - Not writable - always return zeroes.

#### **90 - PCI RECEIVE DMA DATA LENGTH / CONTROL BITS**

Reset Value: xxxxxxxxxxxxxxxxxxxx

This register contains the number of bytes remaining in the current data buffer being filled by the receive DMA. This register is a down counter and is decremented by the number of bytes written each time data is written to the buffer. This register is also used as a temporary holding space for the offset into a frame at which a fragment list based copy begins. The receive DMA control bits may also be read and written through this I/O port.

31 through 20 - Unused.

19 - OWNER: Indicates whether the last descriptor read was owned by the NIC (1) or the host (0). This bit is read only at this location. It may be written through the PRSTAT register.

18 - HEADER: Indicates that this descriptor is for a header copy.

17 - LFFORM: Fragment list format - a '1' indicates that the data length field comes before the pointer in the fragment list. A '0' indicates that the pointer comes before the data length.

16 - FRAGLIST: Indicates that this descriptor points to a fragment list.

15 through 0 - BUFFER LENGTH / OFFSET.

#### **94 - PCI RECEIVE FRAGMENT COUNT**

Reset Value: xxxxxx

This register contains the number of fragments in the current receive DMA fragment list. It is decremented just before each fragment is read.

31 through 6 - Unused.

5 through 0 - FRAGMENT COUNT.

#### **98 - PCI RECEIVE RAM CURRENT ADDRESS**

Reset Value: xxxxxxxxxxxxxxxxxxxx

This register contains the byte address of the data currently being accessed in the local receive RAM.

31 through 18 - Unused.

17 through 0 - RX RAM ADDRESS.

#### **9C - PCI RECEIVE RAM PACKET ADDRESS**

Reset Value: The PRLPAR register will always point to the starting address of the internal receive memory after reset. The actual address will be determined by the memory size jumper settings:

| Size | Reset Value        |
|------|--------------------|
| 128K | 100000000000000000 |
| 32K  | 001000000000000000 |
| 16K  | 000100000000000000 |
| 0    | 000000000000000000 |

This register contains the byte address of the beginning of the frame currently being copied from the local receive RAM. The two LSBs are fixed at zero so that the address will always be DWORD aligned.

31 through 18 - Unused.

17 through 2 - RX RAM ADDRESS.

1 and 0 - Not writable - always return zeroes.

#### **A0 - PCI RECEIVE END OF FRAME ADDRESS**

Reset Value: xxxxxxxxxxxxxxxxxxxx

This register contains the byte address of the DWORD location immediately following the end of the current frame in the local receive RAM. The two LSBs contain the number of valid bytes in the last DWORD of the frame.

31 through 18 - Unused.

17 through 2 - RX RAM ADDRESS.

1 and 0 - ENDING BYTE COUNT.

#### **A4 - PCI RECEIVE DMA STATUS Register**

Reset Value:

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

This register contains the status word that is posted to the receive descriptor chain after a frame has been copied. It includes the status and length of the copied frame as well as the receive DMA status. This register may be read when the host chooses not to copy a frame in receive lookahead mode. Otherwise, it should only be accessed for test purposes.

31 through 16 - RECEIVE FRAME LENGTH:  
Number of bytes in the received frame.

15 - OWNER: Descriptor ownership bit - This bit is writable at this location but may only be read at bit 19 in the PRDLGTH register. When read here, this bit always returns 0 to set descriptor ownership to the host.

14 - HEADER COPIED: Set when the receive status is posted after a header copy. This bit is read only.

13 - FRAGMENT LIST ERROR: Set when all buffers in the fragment list have been filled before the entire receive frame is copied. This bit is read only.

12 - NETWORK STATUS VALID: Set when bits 6 through 0 contain the status from the current frame and bits 31-16 contain the frame length. In the case of a header-copy or fragment-list error, the receive status from the current frame may or may not be posted. In all other cases, this bit will be set. This bit is read only.

11 through 7: Reserved.

6 - RECEIVER DISABLED: This bit is set when the receiver is in monitor mode. Otherwise, returns 0.

5 - BROADCAST ADDRESS RECOGNIZED: This bit is set when a broadcast address has been recognized.

4 - MULTICAST ADDRESS RECOGNIZED: This bit is set when a multicast address (that passes the hash filter) has been recognized.

3 - MISSED PACKET: This bit is set when a packet with a recognized address and without errors (or with masked errors) is not buffered because the device is in monitor mode. This bit is also set when the packet overflows the receive buffer space and cannot be received. Otherwise, returns 0.

2 - CRC ERROR: This bit is set when a frames computed CRC does not match the CRC appended to the frame. If the frame is a runt, this bit will be clear. In MII mode, this bit will also be set if receive error was asserted on the MII interface during reception of the frame.

1 - FRAME ALIGNMENT ERROR: This bit is set if a CRC error has occurred and the frame is not byte aligned.

0 - PACKET RECEIVED INTACT: This bit is set when a packet is received into the buffer space without error.

#### **A8 - RECEIVE RAM BUFFER**

Reset Value:

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

31 through 0 - The receive RAM can be read and written through this I/O port (for test purposes only). The read or write will occur at the address specified in the PRLCAR register. The PRLCAR register will be incremented by four (one DWORD) each time this port is read or written.

#### **AC - RECEIVE MTU CURRENT ADDRESS**

Reset Value: 1000000000000000

This register contains the receive MTU's pointer to the next location it will write in the local receive RAM. This register contains the DWORD address and is write only. Reads to this register return unknown data.

31 through 16 - Unused.

15 through 0 - ADDRESS.

#### **B0 - PCI RECEIVE COPY THRESHOLD Register**

Reset Value: 11111111XX

This register is programmed with the PCI receive copy threshold for the LAN83C171. An early receive warning interrupt will be generated for each frame after the number of bytes, specified in this register, have been copied into the receive data buffers in the host memory. Bits 1 and 0 are ignored; so, the granularity of the threshold is four bytes. The register should only be written at the initialization time.

31 through 11: Unused.

9 through 2: THRESHOLD: For early warning receive interrupt.

1 and 0: Not writable - return unknown data.

#### **B4 - ROMDATA**

Reset Value:

00000000000000000000000000000000

When data is written to the external flash ROM, it is written 8 bits at a time. However, the host will write data to the EPIC 32 bits at a time. The data is written here and stored. It is then accessed 8 bits at a time while the write takes place. Note, however, that this register is accessible for testing purposes only. During normal operation, the host will execute a write to the flash memory by executing a PCI memory write operation to the memory space defined for the ROM.

31 through 0 - ROMDATA: Data is automatically written here when executing a memory write to the ROM address space.

#### **BC - PREEMPTIVE INTERRUPT**

Reset Value: 0000000000

This register is used to set the preemptive interrupt value, the number of bytes before the end of a packet that a packet received interrupt will be issued.

10 through 0 - PREEMPTIVE INTERRUPT VALUE: This value is the number of bytes before the end of the packet that the interrupt will be issued.

#### **C0 - PCI TRANSMIT FIRST DESCRIPTOR ADDRESS**

Reset Value:

xxxxxxxxxxxxxxxxxxxxxxxxxxxx00

This register contains the byte address of the first descriptor for the current transmit packet. It is the location in host memory where the transmit status will be posted when the transmission is completed. The two LSBs are fixed at zero so that the address will always be DWORD aligned. This register is automatically written with the same data as the PTCDAR register whenever a write to that register occurs.

31 through 2 - PCI ADDRESS.

1 and 0 - Not writable - always return zeroes.

#### **C4 - PCI TRANSMIT CURRENT DESCRIPTOR ADDRESS Register**

Reset Value:

XXXXXXXXXXXXXXXXXXXXXXXXXXXX00

This register contains the byte address (in host memory) of the next descriptor that the transmit DMA will read. The two low significant bits are fixed at zero so the address will always be dword aligned. **This register must be initialized once after reset.**

31 through 2: PCI ADDRESS: Address for the next transmit descriptor in the host memory.

1 and 0: Not writable - always return zeroes.

#### **C8 - PCI TRANSMIT HOST DATA ADDRESS**

Reset Value:

XXXXXXXXXXXXXXXXXXXXXXXXXXXX

This register contains the address where the transmit packet data is to be read from host memory. The upper 30 bits are driven onto the PCI Bus as the DWORD address, and are incremented each time a DWORD is read from the host memory. The two LSBs always contain the starting byte address of the data buffer, and are used by the transmit DMA to control byte alignment.

31 through 2 - PCI ADDRESS.

1 and 0 - STARTING BYTE ADDRESS.

#### **CC - PCI TRANSMIT FRAGMENT LIST ADDRESS**

Reset Value:

XXXXXXXXXXXXXXXXXXXXXXXX00

This register contains the current fragment list address. It is the location in host memory of the next fragment list entry that the transmit DMA will read. The two LSBs are fixed at zero so that the address will always be DWORD aligned.

31 through 2 - ADDRESS.

1 and 0 - Not writable - always return zeroes.

#### **D0 - PCI TRANSMIT DMA DATA LENGTH / CONTROL BITS**

Reset Value: xxxxxxxxxxxxxxxxx

This register contains the number of bytes remaining in the current data buffer being read by the transmit DMA. This register is a down counter and is decremented by the number of bytes copied each time data is read from the buffer. The transmit DMA control bits may also be read and written through this I/O port.

31 through 22 - Unused.

21 - OWNER: Indicates whether the last descriptor read was owned by the NIC (1) or the host (0). This bit is read only at this location. It may be written through the PTSTAT register.

20 - LASTDESCR: Indicates that this is the last descriptor for the current transmit frame (Not used when FRAGLIST = 1).

19 - NOCRC: Disable automatic CRC generation for this packet when set.

18 - IAF: When set, interrupt the host after this frame is transmitted.

17 - LFFORM: Fragment list format - a 1 indicates that the data length field comes before

the pointer in the fragment list. A 0 indicates that the pointer comes before the data length.

16 - FRAGLIST: Indicates that this descriptor points to a fragment list.

15 through 0 - BUFFER LENGTH.

#### **D4 - PCI TRANSMIT FRAGMENT COUNT**

Reset Value: XXXXXX

This register contains the number of fragments in the current transmit DMA fragment list. It is decremented just before each fragment is read.

31 through 6 - Unused.

5 through 0 - FRAGMENT COUNT.

#### **D8 - PCI TRANSMIT RAM CURRENT ADDRESS**

Reset Value: 000000000000

This register contains the byte address of the data currently being accessed in the local transmit RAM.

31 through 18 - Unused.

10 through 0 - TX RAM ADDRESS.

#### **DC - EARLY TRANSMIT THRESHOLD Register**

Reset Value: XXXXXXXXXXXX

This register is programmed with the early transmit threshold for the LAN83C171. Transmission will begin on the network after the number of bytes, specified in this register, have been loaded into the local transmit RAM. Bits 1 and 0 are ignored; so, the granularity of the threshold is four bytes. Data written into this register will automatically be stored in the early transmit count register at the same time. The register should only be written at the initialization time.

31 through 11: Unused.

10 through 2: THRESHOLD: Early transmit.

1 and 0: Not writable - return unknown data.

#### **E0 - PCI EARLY TRANSMIT COUNT**

Reset Value: xxxxxxxxx

This counter contains the number of bytes to be copied into the local transmit buffer before the early transmit threshold is reached. The counter is loaded with the early transmit threshold value at the beginning of each frame and counts down to zero. This register is automatically written with the same data as the ETXTHR register whenever a write to that register occurs.

31 through 11 - Unused.

10 through 2 - EARLY TRANSMIT COUNT.

1 and 0 - Not writable - return unknown data.

#### **E4 - PCI TRANSMIT DMA STATUS**

Reset Value: xxxxxxxxxxxxxx

This register contains a copy of the transmit status from the most recently completed transmission. The value is stored in this register until it can be posted to the transmit descriptor chain. Data from the host may not be written in to this register. When this register is written by the host, it will be loaded with the current value in the TXSTAT register. Reads work normally.

The transmit length register and transmit length counter are also writable through the upper word at this address.

31 through 16 - TRANSMIT LENGTH: When this register is written, these bits are stored into both the transmit length register and the transmit length counter. These bits are not readable, and return unknown data when read.

15 - OWNER: Descriptor ownership bit - This bit is writable at this location, but may only be read at bit 21 in the PTDLGTH register. When read here, this bit always returns a '0' to set the descriptor ownership to the host.

14 and 13 - Unused.

12 through 8 - COLLISION COUNT: These bits contain the number of collisions detected while attempting to transmit the current packet. Bit 12 also indicates if a transmit abort for excessive collisions occurred.

7 - DEFERRING: This bit is set when the interframe gap state machine is deferring. If the PHY has asserted the collision line as a result of jabber, this bit will stay set indicating the jabber condition.

6 - OUT OF WINDOW COLLISION: This bit is set if a collision is detected more than one slot time after the start of a transmission. The transmission is aborted under these conditions.

5 - COLLISION DETECT HEARTBEAT: This bit is set to a '1' during the transmission of each packet. It is set to a '0' if a collision is detected within 36 bit times of the end of each packet transmission. If no collision is detected within this window, it remains a '1'. This bit always returns a '0' during full duplex mode.

4 - UNDERRUN: This bit is set when the transmit DMA is unable to supply the transmitter with enough data to maintain the frame transmission.

3 - CARRIER SENSE LOST: This bit is set if the carrier is lost during a packet transmission. Carrier sense is monitored from its rising edge at the start of the outgoing frame's echo. The transmission is not aborted upon the loss of the carrier. This bit will always return a '0' during full duplex mode.

2 - TRANSMITTED WITH COLLISIONS: When set, this bit indicates that the frame collided at least once with another frame on the network. It is not set for either out-of-window collisions or excessive collision aborts.

1 - NON-DEFERRED TRANSMISSION: This bit is set if the frame was transmitted successfully without deferring. A deferred transmission can only occur the first time an attempt is made to send a packet. Collisions are not deferred transmissions.

0 - PACKET TRANSMITTED: This bit is set to indicate the transmission of a packet without excessive collisions or an abort.

## **E8 - TRANSMIT RAM BUFFER**

Reset Value:

xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

31 through 0 - The transmit RAM can be read and written through this I/O port (for test purposes only). The read or write will occur at the address specified in the PTLCAR register. The PTLCAR register will be incremented by four (one DWORD) each time this port is read or written.

## **EC - PCI TRANSMIT 2 FIRST DESCRIPTOR ADDRESS**

Reset Value:

xxxxxxxxxxxxxxxxxxxxxxxxxxxx00

If two frames are loaded into the local transmit RAM, this register contains the byte address of the first descriptor for the second transmit packet. It is the location in host memory where the transmit status will be posted when the transmission of that frame is completed. The two LSBs are fixed at zero so that the address will always be DWORD aligned.

31 through 2 - PCI ADDRESS.

1 and 0 - Not writable - always return zeroes.

**FO – FEVTR Register**

Reset Value: 0XXXXXXXXXXXXXX

This bit is used for CardBus purposes. CardBus mode must be set in the NVCTL register bit 6 for this function to be available.

15 - FEVTR: This function event register bit is a register for holding interrupts in the CardBus environment.

14 through 0: Unused.

**F4 – FEVTRMSKR Register**

Reset Value: 0XXXXXXXXXXXXXX

This bit is the mask for the fevtr bit in CardBus operations. CardBus mode must be set in the NVCTL register bit 6 for this function to be available.

15 - FEVTRMSKR: This bit is used for holding the mask for the fevtr interrupt. If it is '1', an interrupt derived from fevtr will not go out to the host.

14 through 0: Unused.

**F8 – FPRSTR Register**

Reset Value: 01

This register is the function present register for CardBus operations. CardBus mode must be set in the NVCTL register bit 6 for this function to be available.

1 - FPRST\_INT: This bit reflects the interrupt status before it has been sent to the function event register.

0 - The CardBus bus functions are always present, and this will always return a '1', and it can not be written.

**FC – FFRCEVTR Register**

Reset Value: 0XXXXXXXXXXXXXX

This register is the function force event register for CardBus operations. CardBus mode must be set in the NVCTL register bit 6 for this function to be available.

15 - FFRCEVTR: When this bit is written to, the fevtr register will become '1'.

14 through 0: Unused.

## PCI CONFIGURATION REGISTERS BITS DESCRIPTION

### 00 - DEVICE ID/VENDOR ID Register

31 through 16 - DEVICE ID: This read only field returns the LAN83C171 device ID (0005h). Bit 31, when 1, indicates the Component Products Division and when 0, indicates the System Products division. The remaining bits are assigned arbitrarily to identify each System Products Division PCI device uniquely.

15 through 0 - VENDOR ID: This read only field returns the SMSC Vendor ID (10B8h).

### 04 - PCI STATUS/COMMAND Register

#### Status Register

Reset Value: 0000000010000000

Bits in this register are set internally by the LAN83C171. Bits are cleared by writing a 1 to their respective locations. Writing 0 to a bit has no effect (in register test mode, writing 0 sets the bit).

31 - DETECTED PARITY ERROR: This bit is set whenever the LAN83C171 detects a parity error, even if parity error handling is disabled.

30 - SIGNALLED SYSTEM ERROR: This bit is set whenever the LAN83C171 asserts system error. The LAN83C171 asserts system error when an address parity error is detected and both 'nSERR enable' and 'Parity Error Response' bits are set.

29 - RECEIVED MASTER ABORT: This bit is set whenever an LAN83C171 bus master transaction is terminated with master-abort.

28 - RECEIVED TARGET ABORT: This bit is set whenever an LAN83C171 bus master transaction is terminated with target-abort.

27 - SIGNALLED TARGET ABORT: This bit is not implemented because the LAN83C171 never signals target abort (always returns 0).

26 and 25 - DEVSEL TIMING: These two read only bits always return "00" to indicate that the LAN83C171 always asserts DEVSEL with fast timing (zero wait states).

24 - DATA PARITY DETECTED: This bit is set whenever the following three conditions are met: 1) the LAN83C171 is acting as bus master on the PCI bus; 2) the LAN83C171 asserts nPERR or observes nPERR asserted; 3) the Parity Error Response bit is set.

23 - FAST BACK-TO-BACK CAPABLE: This read only bit always returns 1 to indicate that the LAN83C171 is capable of accepting fast back-to-back transactions when the transactions are not to the same agent.

22 - UDF SUPPORTED: This read only bit tells the host system whether or not the LAN83C171 supports user definable features. The value of this bit is recalled from EEPROM at power up and stored in the NVCTL register. This bit should be programmed to zero in the EEPROM to indicate that the LAN83C171 does not support user definable features.

21 - 66 MHz CAPABLE: This bit always returns zero to indicate that the LAN83C171 is not 66 MHz capable.

20 through 16: Reserved (always return 0).

#### COMMAND Register

Reset Value: 0000000000000000

15 through 10: Reserved (always return 0).

9 - FAST BACK-TO-BACK ENABLE: This bit is not implemented as the LAN83C171 never

performs bus master transactions to two different devices (always returns 0).

8 - nSERR ENABLE: When this bit is set the LAN83C171 may assert nSERR. When this bit is cleared nSERR signaling is disabled.

7 - WAIT CYCLE CONTROL: This bit is not implemented as the LAN83C171 does not employ address/data stepping (always returns 0).

6 - PARITY ERROR RESPONSE: When this bit is set, the LAN83C171 responds to parity errors. When cleared, the LAN83C171 ignores parity errors.

5 through 3 - VGA PALETTE SNOOP, MEMORY WRITE AND INVALIDATE ENABLE, SPECIAL CYCLES: Not implemented (always return 0).

2 - BUS MASTER ENABLE: The LAN83C171 may only act as bus master on the PCI bus when this bit is set. When this bit is cleared, the LAN83C171 disables its PCI request signal.

1 - MEMORY SPACE ENABLE: The LAN83C171 may respond to memory space accesses when this bit is set. When the bit is cleared, the LAN83C171 does not respond to memory space accesses.

0 - I/O SPACE ENABLE: The LAN83C171 may respond to I/O space accesses when this bit is set. When the bit is cleared, the LAN83C171 does not respond to I/O space accesses.

#### **CLASS CODE/REVISION ID**

31 through 24 - BASE CLASS: This read only field returns the Network Controller Base Class (02h).

23 through 16 - SUB CLASS: This read only field returns the Ethernet Controller Sub-Class (00h).

15 through 8 - PROGRAMMING INTERFACE: This read only field returns 00h (no specific register-level programming interface defined).

7 through 0 - REVISION ID: This read only field returns the LAN83C171 silicon revision ID 06h

#### **0C - HEADER TYPE/LATENCY TIMER Register**

31 THROUGH 24: Unused (returns 00h).

23 - MULTI-FUNCTION DEVICE: This bit returns 0 to indicate that the LAN83C171 is a single function PCI device.

22 through 16 - HEADER TYPE: Specifies the format of bytes 10h - 3Ch in the configuration space (00h).

15 through 8 - LATENCY TIMER: This byte is programmed with the value of the Latency Timer (in PCI bus clocks) for LAN83C171 bus master operations. The bottom three bits are hardwired to 0, giving the latency timer a granularity of 8 clocks. This register is 00h after reset.

7 through 0: Unused (returns 00h).

#### **10 - I/O BASE ADDRESS Register**

Reset Value:

XXXXXXXXXXXXXXXXXXXXXXXXX00000001

31 through 8: BASE ADDRESS

7 through 2: Return zeroes to indicate that 256 bytes of address space are required.

1: Reserved (always returns zero).

0 - I/O SPACE INDICATOR: This read only bit returns a 1 to map the control registers into I/O space.

#### **14 - MEMORY BASE ADDRESS Register**

Reset Value:

XXXXXXXXXXXXXXXXXXXXXXXXX000000000000

31 through 12 - BASE ADDRESS: When the memory map enable bit in the NVCTL register is set to 1, these are read/write register bits. When memory mapping is disabled, these bits return zeroes.

11 through 4: Return zeroes to indicate that 4 Kbytes of address space are required.

3 - PREFETCHABLE: This read only bit returns 0 to indicate that this address space is not prefetchable.

2 and 1 - TYPE: These two read only bits return "00" to indicate that the control registers may be mapped anywhere in the 32 bit address space.

0 - MEMORY SPACE INDICATOR: This read only bit returns a 0 to map the control registers into memory space.

#### **28 - CIS POINTER Register**

Reset Value:

01110000000000000000000010000000

This read only register points to the location of the CardBus "Card Information Structure (CIS)."

31: Reserved (always returns zero).

30 through 28 - ADDRESS SPACE INDICATOR: Returns 7h to indicate that the CIS is mapped into expansion ROM space.

27 through 24 - ROM IMAGE NUMBER: Returns 0h to indicate that the CIS is located in image 0 in the expansion ROM.

23 through 0 - OFFSET: Returns 000100h to indicate that the CIS starts 256 bytes into image 0 of the expansion ROM.

#### **2C - SUBSYSTEM ID/SUBSYSTEM VENDOR ID Register**

Reset Value:

0XXXXXXXXXXXXXXXXX0000000000000000

This read only register is used to uniquely identify the add-in board or subsystem with LAN83C171 residing on it. The value is recalled from EEPROM after power-up reset.

31 through 16 - SUBSYSTEM ID: This field is vendor specific and may be assigned freely. Bit 31 is hardwired to 0. The remaining bits are recalled from EEPROM.

15 through 0 - SUBSYSTEM VENDOR ID: This field identifies the subsystem vendor. The value is assigned by the PCI SIG.

#### **30 - EXPANSION ROM BASE ADDRESS Register**

Reset Value:

XXXXXXXXXXXXXXXXX0000000000000000

31 through 16: BASE ADDRESS

15 through 1: Return zeroes to indicate that 64 Kbytes of address space are required.

1 - ADDRESS DECODE ENABLE: Controls whether or not the LAN83C171 accepts accesses to its expansion ROM.

#### **3C - BUS REQUIREMENTS/INTERRUPT MAP Register**

Reset Value:

0000000000000000000000001XXXXXXXXX

31 through 24 - MAXIMUM LATENCY: This read only field specifies how often the LAN83C171 needs to gain access to the PCI bus.\* To enable performance tuning, the value is recalled from EEPROM after reset.

23 through 16 - MINIMUM GRANT: This read only field specifies the length of a burst period the LAN83C171 needs assuming a 33 MHz PCI

clock rate.\* To enable performance tuning, the value is recalled from EEPROM after reset.

15 through 8 - INTERRUPT PIN: This read only field returns 01h to indicate that the LAN83C171's interrupt output is connected to the nINTA pin on the PCI connector.

7 through 0 - INTERRUPT LINE: This byte is programmed with interrupt routing information. The value indicates the input of the system's interrupt controller(s) that LAN83C171's interrupt pin is connected to. Values in this register are system architecture specific.

\* The maximum latency and minimum grant registers are used to indicate the LAN83C171's desired settings for Latency Timer values. Both registers specify a period of time in units of 1/4 microsecond. For example, if the LAN83C171 needs to perform a 2 microseconds long burst every 11 microseconds (on the average), then the maximum latency register would read 44 (1Ch) and the minimum grant register would read 8 (08h).

## **DC – POWER MANAGEMENT REGISTER BLOCK (1<sup>st</sup> Half)**

Reset Value:

x1110110001x00010000000000000001

Bits 31 through 16 of this register are collectively known as the “Power Management Capabilities”. This entire register is read only.

31 - D3(COLD1) PME SUPPORT: This bit indicates whether the EPIC can generate a PME during power state D3(Cold1) based on the capabilities of the board on which it is implemented. To determine this, the EPIC will latch the state of I/O pin MD(5) during initial power-up. If the board is capable of supplying auxiliary power to the EPIC upon the removal of PCI Bus power, then this bit should be pulled to a high(1). Otherwise, it should be pulled to a low(0).

30 through 27 - D3(HOT) THROUGH D0 PME SUPPORT: These bits will read back as “1110” to indicate that the EPIC can generate a PME during power states D3(Hot), D2, and D1, but not power state D0.

26 through 25 - D2 AND D1 SUPPORT: These bits will read back as “11” to indicate that the EPIC supports power states D1 and D2.

24 through 22 - Reserved. Will read back as “000.”

21 - DEVICE SPECIFIC INITIALIZATION: This bit will read back as a ‘1’ to indicate that the EPIC requires device specific initialization (beyond the standard PCI configuration header) prior to being usable as a NIC.

20 - AUXILIARY POWER SOURCE: This bit will read back as a ‘1’ to indicate that the EPIC requires auxiliary power during power state D3(Cold1) in order to generate a PME. This is true only if the board on which the EPIC is currently implemented supports PME generation during D3 (Cold1) as indicated by bit 31 of this register. If, however, the board on which the EPIC is currently implemented does not support PME generation during D3 (Cold1), then bit 20 will read back as zero(0).

19 - PME CLOCK: This bit will read back as a zero(0) to indicate that the EPIC does not rely on the presence of the PCI Bus clock to generate a PME.

18 through 16 - VERSION SUPPORT: These bits will read back as “001” to indicate that the EPICXF implements the registers as described in revision 1.0 of the PCI power management specification.

15 through 8 - NEXT CAPABILITY POINTER: These bits contain the pointer address for the next capability within the capabilities list. Will read back as 00h to indicate that there are no other capabilities available for the EPIC.

7 through 0 - CAPABILITY ID: These bits contain the “Capability Identifier” code. The code for the “Power Management Register Block” is 01h.

## **E0 – POWER MANAGEMENT REGISTER BLOCK (2<sup>nd</sup> Half)**

Reset Value:

00000000000000000000000000000000

Bits 23 through 0 of this register are collectively referred to as the “Power Management Control/Status Register.”

31 through 24 - PM DATA: These bits will return 00h (read only) since the EPIC does not support the read back of power and heat dissipation information.

23 through 16 - BRIDGE SUPPORT EXTENSIONS: These bits will return 00h (read only) since the EPIC is not a PCI-to-PCI bridge.

15 - PME STATUS: This bit will indicate if the EPIC has detected an enabled PME while in one of the low power states regardless of whether it is enabled to notify the OS. This bit can be cleared by writing a '1' to it. Writing a '0' to this location has no effect.

14 and 13 - PM DATA SCALE: These bits will return "00" (read only) since the EPIC does not support the read back of power and heat dissipation information.

12 through 9 - PM DATA SELECT: These bits will return "0000" (read only) since the EPIC does not support the read back of power and heat dissipation information.

8 - PME ENABLE: This bit enables the EPIC to notify the OS of the occurrence of a PME.

7 through 2 - Reserved. Will read back as "000000" (read only).

1 and 0 - POWER STATE: These bits allow the driver to specify and query the power state of the EPIC. D0 corresponds to 00b, D1 to 01b, D2 to 10b, and D3 to 11b.

## OPERATIONAL DESCRIPTION

### Maximum Guaranteed Ratings

Operating Temperature Range..... 0°C To +70°C  
 Storage Temperature Range..... -55°C To +150°C  
 Lead Temperature Range (soldering, 10 Seconds)..... +325°C  
 Positive Voltage on any pin with respect to Ground.....  $V_{DD} + 0.3V$   
 Negative Voltage on any pin with respect to Ground..... -0.3V  
 Maximum  $V_{DD}$ ..... +TBDV

The above is a stress rating only. Stresses greater than the ones listed above could damage the device permanently. Device operation outside of the above stated conditions is not recommended.

Note: When powering this device from a laboratory or system power supply, it is important that the Absolute Maximum Ratings are not exceeded. Some power supplies exhibit voltage spikes on their outputs when the AC power is switched on or off. Also, voltage transients on the AC power line may appear on the DC outputs. If such possibilities exist, a clamping circuit should be used.

### DC Electrical Characteristics

( $T_A = 0^\circ\text{C} \text{ --- } +70^\circ\text{C}$ ;  $V_{DD} = +4.7V \text{ --- } +5.3V$ )

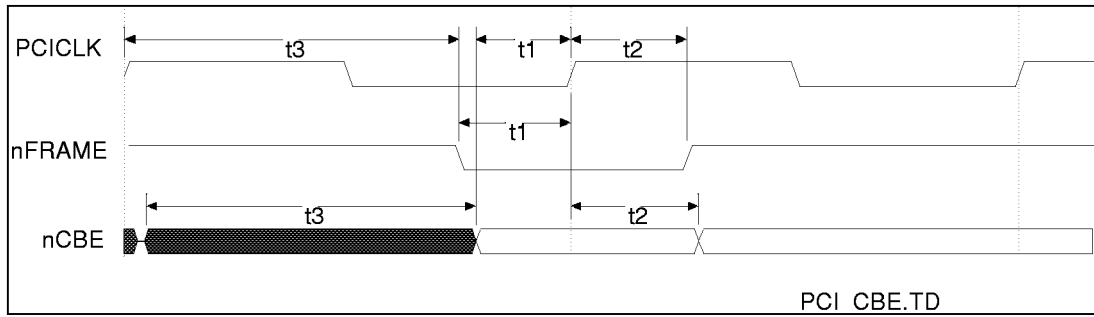
| PARAMETER                                                                  | SYMBOL     | MIN | TYP | MAX | UNITS         | COMMENTS                    |
|----------------------------------------------------------------------------|------------|-----|-----|-----|---------------|-----------------------------|
| Current Supply -Power Up                                                   | $I_{CCP}$  |     | TBD |     | mA            | Power Up in Power Down mode |
| Current Supply - Active                                                    | $I_{CC}$   |     | TBD | TBD | mA            |                             |
| Current Supply - Idle                                                      | $I_{CSBY}$ |     | TBD |     | mA            | No Network Activity         |
| <b>Input Current Leakage - dc_Ik1</b>                                      |            |     |     |     |               |                             |
| Low Input Leakage                                                          | $I_{IL}$   | -10 |     | +10 | $\mu\text{A}$ | $V_{IN} = 0.5V$             |
| High Input Leakage                                                         | $I_{IH}$   | -10 |     | +10 | $\mu\text{A}$ | $V_{IN} = 2.7V$             |
| <b>Output Current Leakage - dc_Ik2</b>                                     |            |     |     |     |               |                             |
| Low Output Leakage                                                         | $I_{OL}$   | -10 |     | +10 | $\mu\text{A}$ | $V_{IN} = 0.0V$             |
| High Output Leakage<br>(Buffer Types: TBS -for buffers w/o internal Pulls) | $I_{OH}$   | -10 |     | +10 | $\mu\text{A}$ | $V_{IN} = 5.3V$             |
| <b>PCI Clock - <math>I_{PCLK}</math></b>                                   |            |     |     |     |               |                             |
| Low Input Voltage                                                          | $V_{IL}$   |     |     | 0.8 | V             |                             |
| High Input Voltage                                                         | $V_{IH}$   | 2.2 |     |     | V             |                             |
| <b>System Clock - <math>I_{SCLK}</math></b>                                |            |     |     |     |               |                             |
| Low Input Voltage                                                          | $V_{IL}$   |     |     | 0.8 | V             |                             |
| High Input Voltage                                                         | $V_{IH}$   | 2.6 |     |     | V             |                             |

| PARAMETER                                                                                | SYMBOL               | MIN | TYP | MAX  | UNITS  | COMMENTS                                                       |
|------------------------------------------------------------------------------------------|----------------------|-----|-----|------|--------|----------------------------------------------------------------|
| $I_{PCI}, I_{TTL}, I_{O_{PCI}}, I_{O_{TTLX}}$<br>Low Input Voltage<br>High Input Voltage | $V_{IL}$<br>$V_{IH}$ | 2.0 |     | 0.8  | V<br>V |                                                                |
| $I_{AN}$<br>Low Input Voltage<br>High Input Voltage                                      | $V_{IL}$<br>$V_{IH}$ | 2.5 |     | 0.0  | V<br>V | $I_{IL} = TBDmA$<br>$I_{IH} = TBDmA$                           |
| $O_{TTLX}, I_{O_{TTLX}}$<br>Low Output Voltage<br>High Output Voltage                    | $V_{OL}$<br>$V_{OH}$ | 2.4 |     | 0.4  | V<br>V | X for $I_L = 2, 4, \& 8mA$<br>loads for the respective<br>pins |
| $O_{PCI}, I_{O_{PCI}}$<br>Low Output Voltage<br>High Output Voltage                      | $V_{OL}$<br>$V_{OH}$ | 2.5 |     | 0.55 | V<br>V | $I_L = 6mA$                                                    |

**Capacitance** ( $T_A = 25^\circ C$ ;  $f_C = 1MHz$ ;  $V_{DD} = +5V$ )

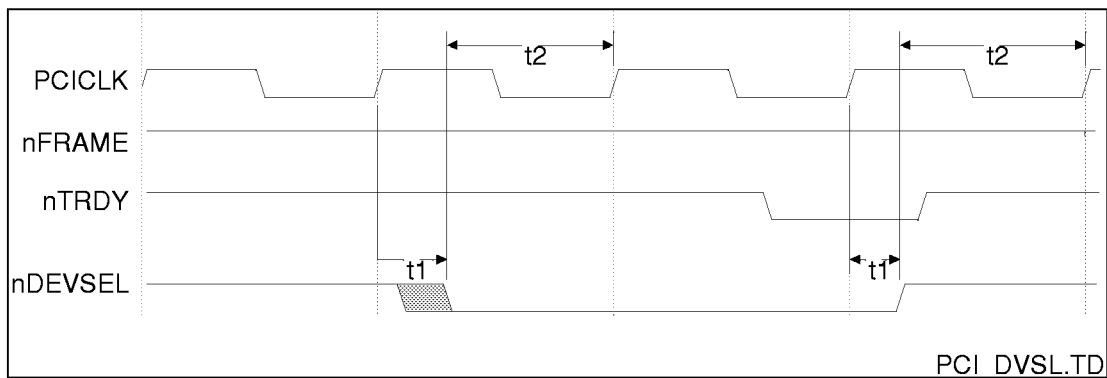
| PARAMETER                      | SYMBOL       | MIN | TYP | MAX        | UNITS | COMMENTS                                                                             |
|--------------------------------|--------------|-----|-----|------------|-------|--------------------------------------------------------------------------------------|
| PCI Clock Input<br>Capacitance | $C_{INPCLK}$ |     |     | <i>TBD</i> | pF    | The pin under test tied<br>to AC ground. All other<br>pins tied to <i>TBD</i> ground |
| SYS Clock Input<br>Capacitance | $C_{INSCLK}$ |     |     | <i>TBD</i> | pF    |                                                                                      |
| Input Capacitance              | $C_{IN}$     |     |     | <i>TBD</i> | pF    |                                                                                      |
| Output Capacitance             | $C_{OUT}$    |     |     | <i>TBD</i> | pF    |                                                                                      |

## TIMING DIAGRAMS



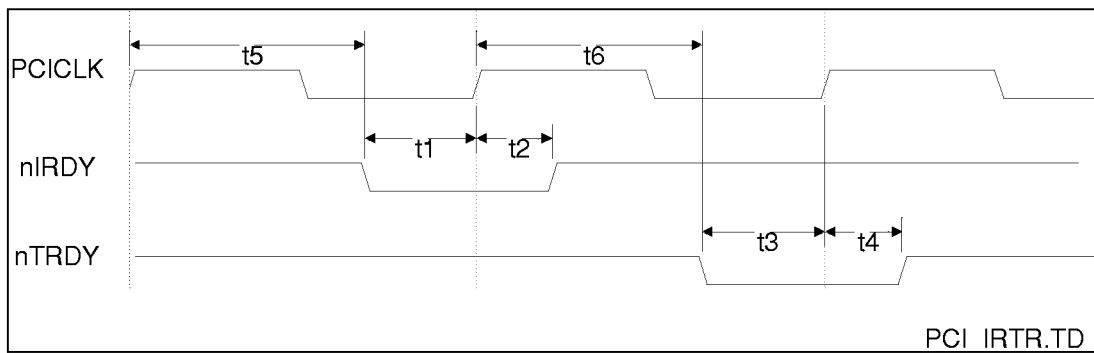
**FIGURE 6 - PCI COMMAND TIMING**

| NAME | MASTER |      | TARGET |     | DESCRIPTION                                                                 |
|------|--------|------|--------|-----|-----------------------------------------------------------------------------|
|      | MIN    | MAX  | MIN    | MAX |                                                                             |
| t1   |        |      | 7ns    |     | Input setup to clock                                                        |
| t2   | 2ns    | 11ns | 0ns    |     | (Master) Clock to signal valid delay<br>(Target) Input hold time from clock |
| t3   | 2ns    | 11ns |        |     | Clock to signal valid delay                                                 |



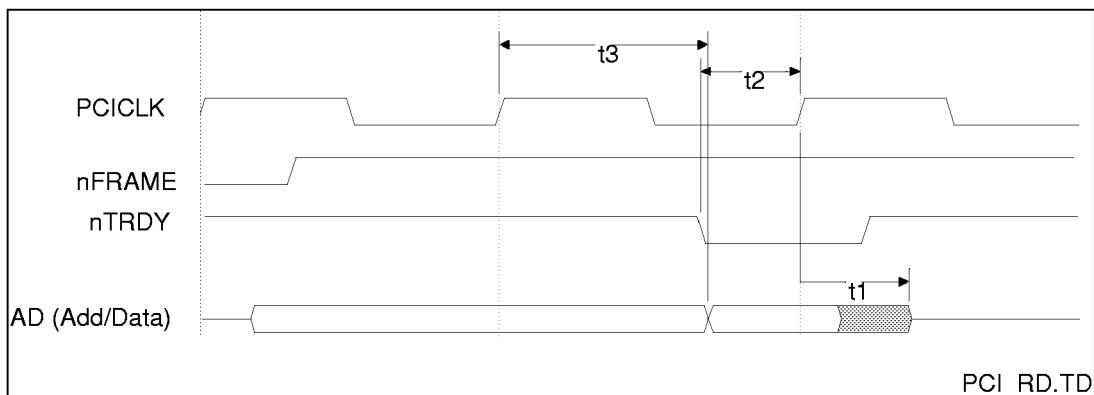
**FIGURE 7 - PCI/nDEVSEL TIMING**

| NAME | MASTER |     | TARGET |      | DESCRIPTION                                                                 |
|------|--------|-----|--------|------|-----------------------------------------------------------------------------|
|      | MIN    | MAX | MIN    | MAX  |                                                                             |
| t1   | 0ns    |     | 2ns    | 11ns | (Master) Input hold time from clock<br>(Target) Clock to signal valid delay |
| t2   | 7ns    |     |        |      | Input setup to clock                                                        |



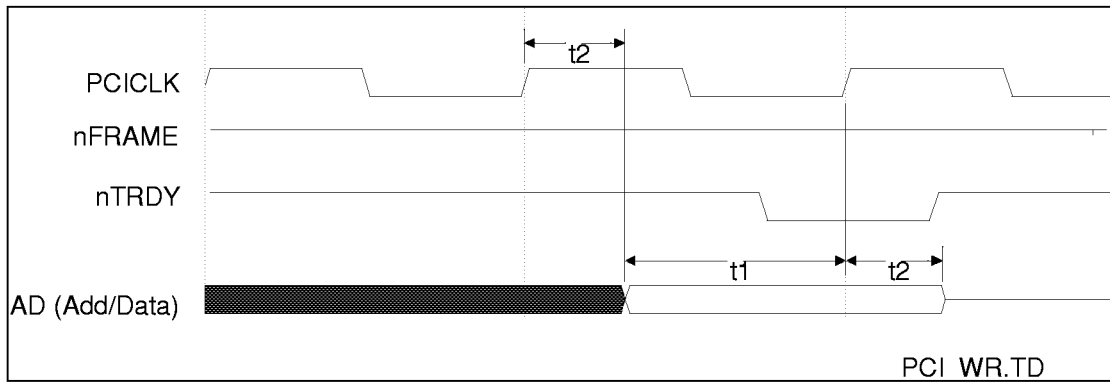
**FIGURE 8 - PCI/nIRDY AND nTRDY TIMING**

| NAME | MASTER |      | TARGET |      | DESCRIPTION                                                                 |
|------|--------|------|--------|------|-----------------------------------------------------------------------------|
|      | MIN    | MAX  | MIN    | MAX  |                                                                             |
| t1   |        |      | 7ns    |      | Input setup time to clock                                                   |
| t2   | 2ns    | 11ns | 0ns    |      | (Master) Clock to signal valid delay<br>(Target) Input hold time from clock |
| t3   | 7ns    |      |        |      | Input setup time to clock                                                   |
| t4   | 0ns    |      | 2ns    | 11ns | (Master) Input hold time from clock<br>(Target) Clock to signal valid delay |
| t5   | 2ns    | 11ns |        |      | Clock to signal valid delay                                                 |
| t6   |        |      | 2ns    | 11ns | Clock to signal valid delay                                                 |



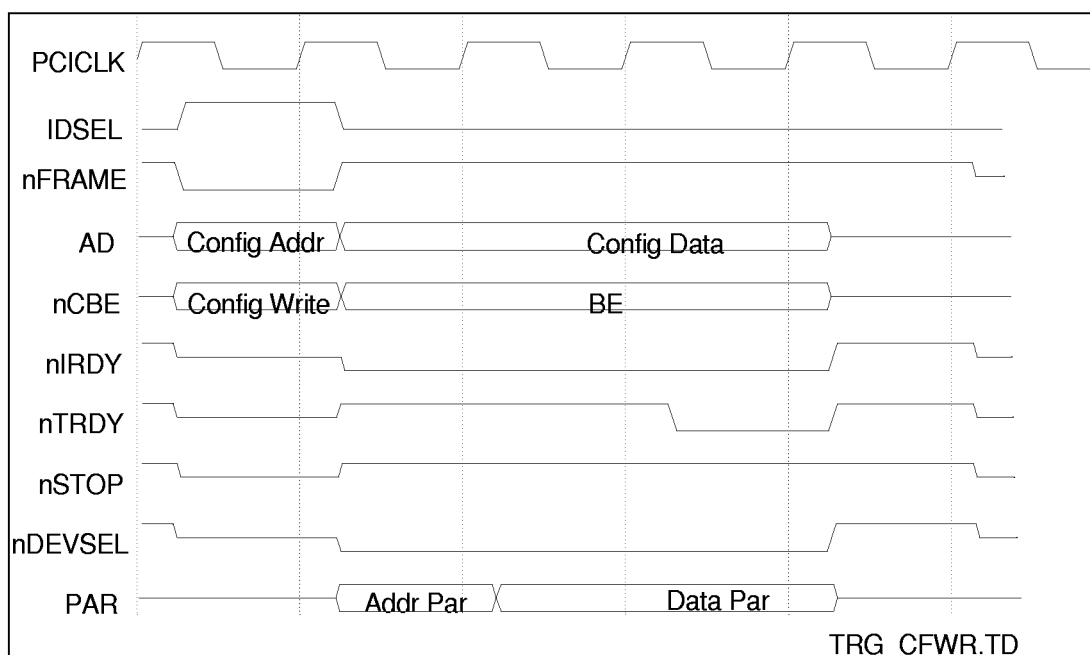
**FIGURE 9 - PCI/DATA READ TIMING**

| NAME | MASTER |     | TARGET |      | DESCRIPTION                                                           |
|------|--------|-----|--------|------|-----------------------------------------------------------------------|
|      | MIN    | MAX | MIN    | MAX  |                                                                       |
| t1   | 0ns    |     | 2ns    | 11ns | (Master) Input hold time from clock<br>(Target) Clock to signal valid |
| t2   | 7ns    |     |        |      | Input setup time to clock                                             |
| t3   |        |     | 2ns    | 11ns | Clock to signal valid                                                 |



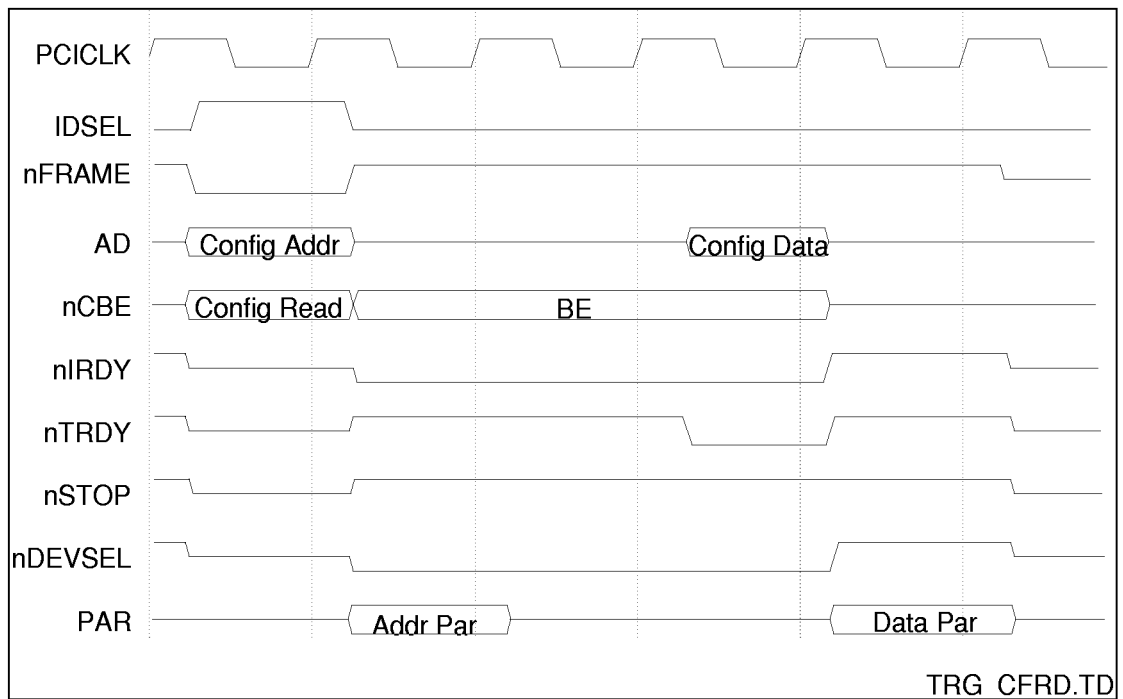
**FIGURE 10 - PCI/DATA WRITE TIMING**

| NAME | MASTER |      | TARGET |     | DESCRIPTION                                                                 |
|------|--------|------|--------|-----|-----------------------------------------------------------------------------|
|      | MIN    | MAX  | MIN    | MAX |                                                                             |
| t1   |        |      | 7ns    |     | Input setup time to clock                                                   |
| t2   | 2ns    | 11ns | 0ns    |     | (Master) Clock to signal valid delay<br>(Target) Input hold time from clock |



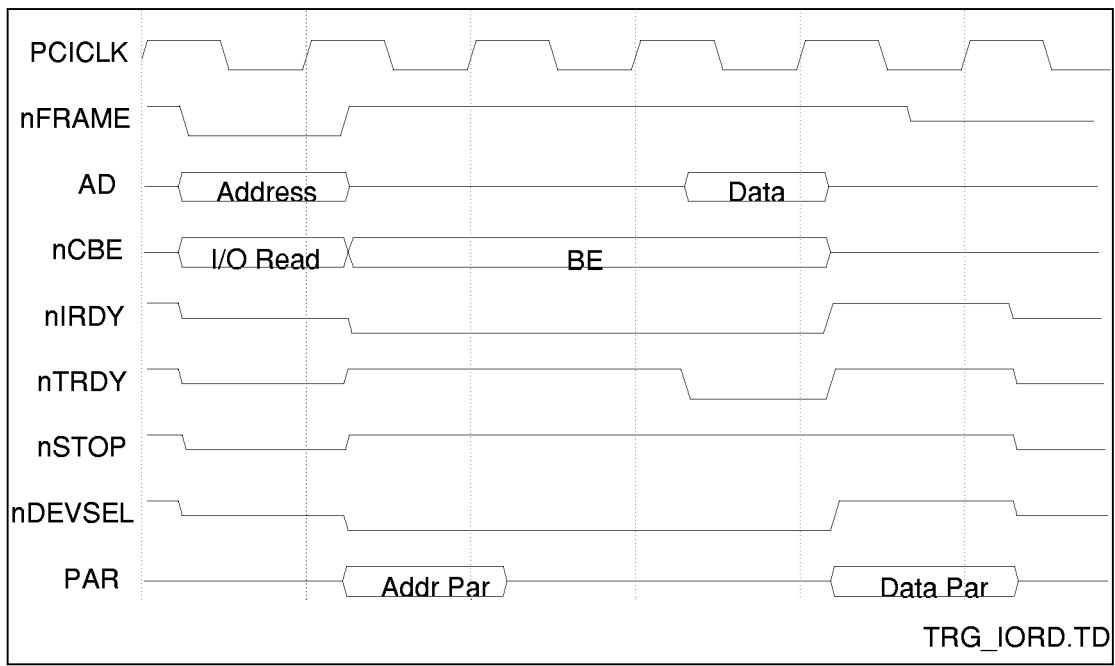
**FIGURE 11 - PCI - TYPICAL CONFIGURATION WRITE/EPIC IS TARGET**

**PCI Bus Cycle Illustration**



**FIGURE 12 - PCI - TYPICAL CONFIGURATION READ/EPIC IS TARGET**

**PCI Bus Cycle Illustration**



**FIGURE 13 - PCI - TYPICAL I/O READ/EPIC IS TARGET**

**PCI Bus Cycle Illustration**

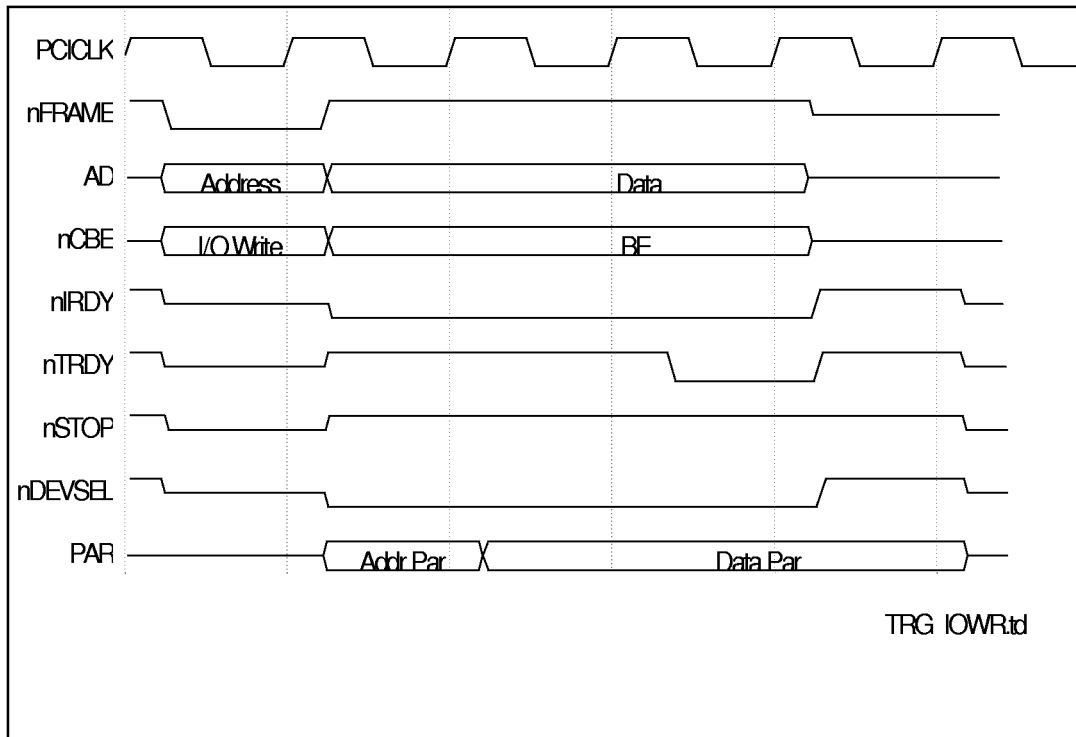
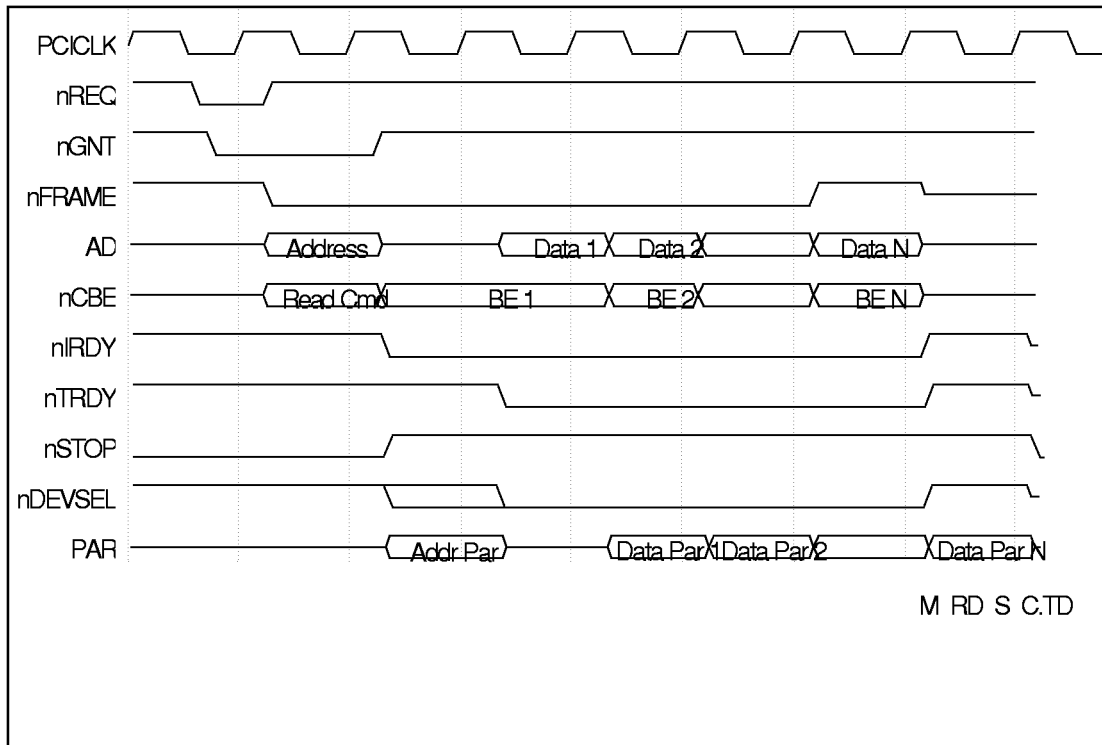


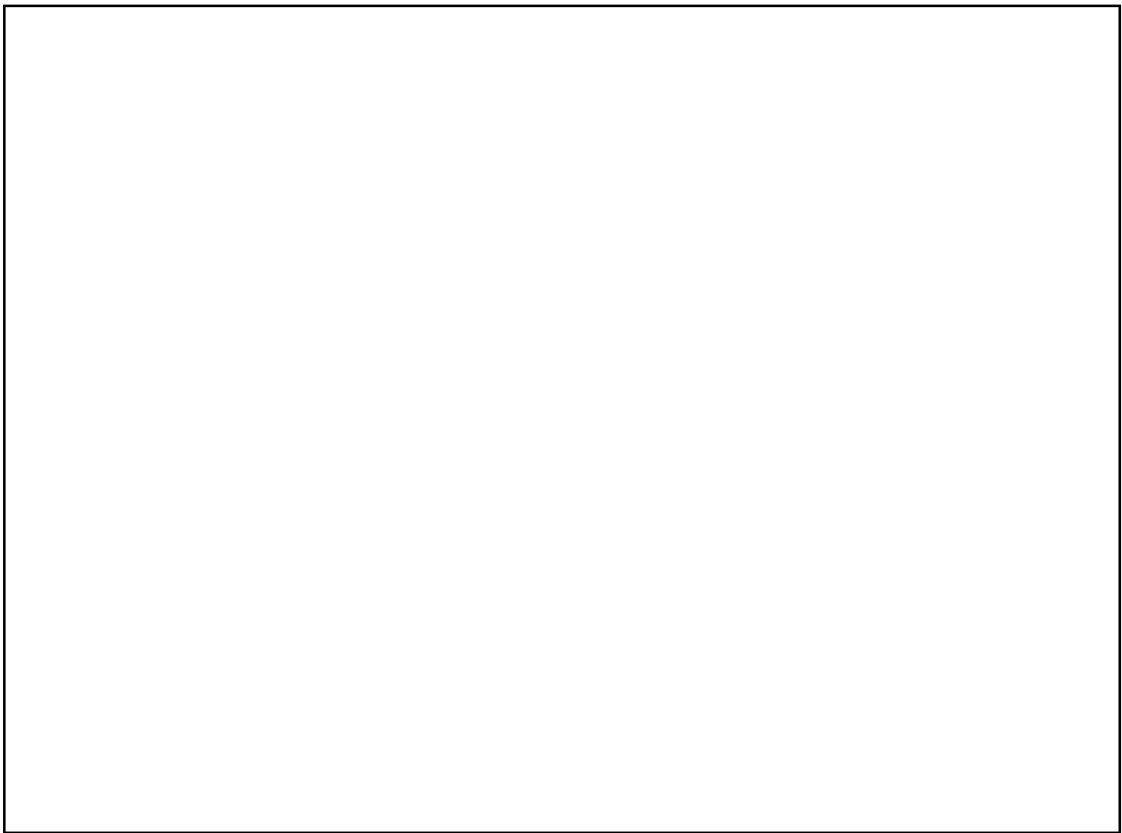
FIGURE 14 - PCI - TYPICAL I/O WRITE/EPIC IS TARGET

PCI Bus Cycle Illustration



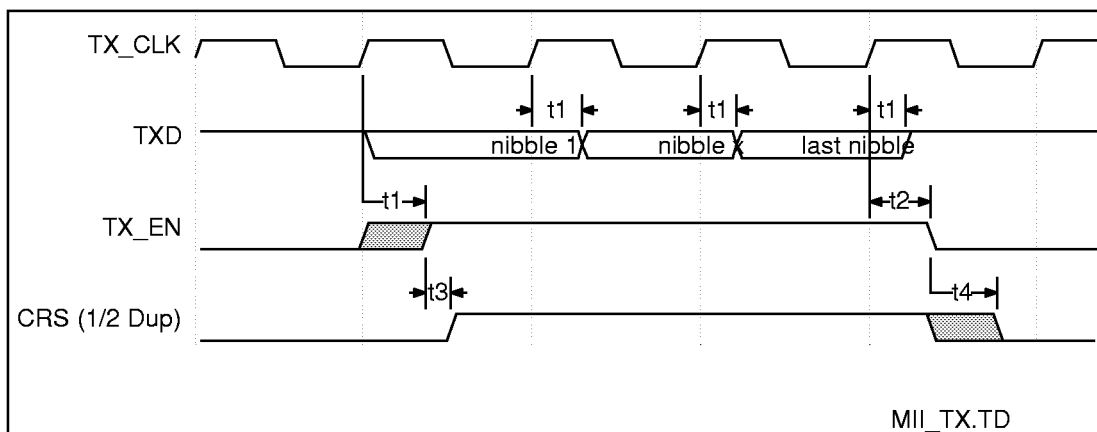
**FIGURE 15 - PCI - TYPICAL READ TRANSACTION/SYSTEM MEMORY TO CHIP  
EPIC IS BUS MASTER**

**PCI Bus Cycle Illustration**



**FIGURE 16 - PCI - TYPICAL WRITE TRANSACTION/CHIP TO SYSTEM MEMORY  
EPIC IS BUS MASTER**

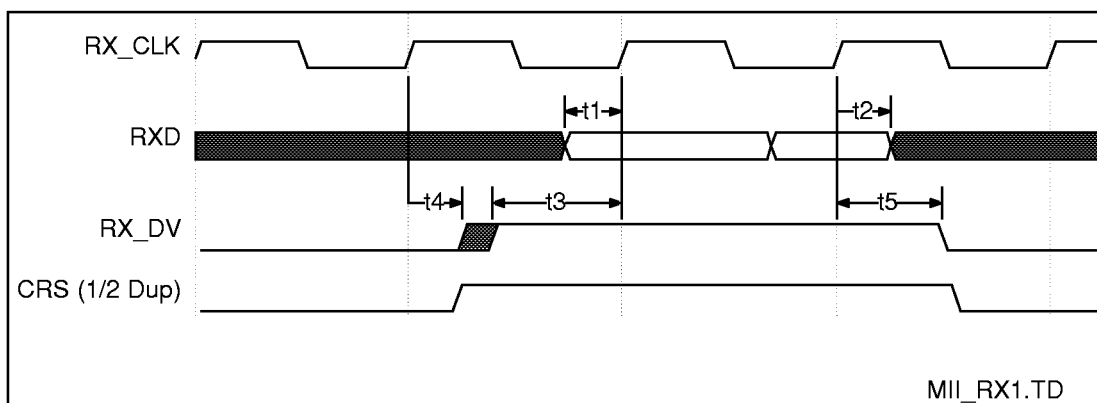
**PCI Bus Cycle Illustration**



**FIGURE 17 - MII - TRANSMIT TIMING FOR 10/100Mb/s**

Note: Clock Frequency Changes to 2.5 MHz for 10 Mb/s Nibble Transfers

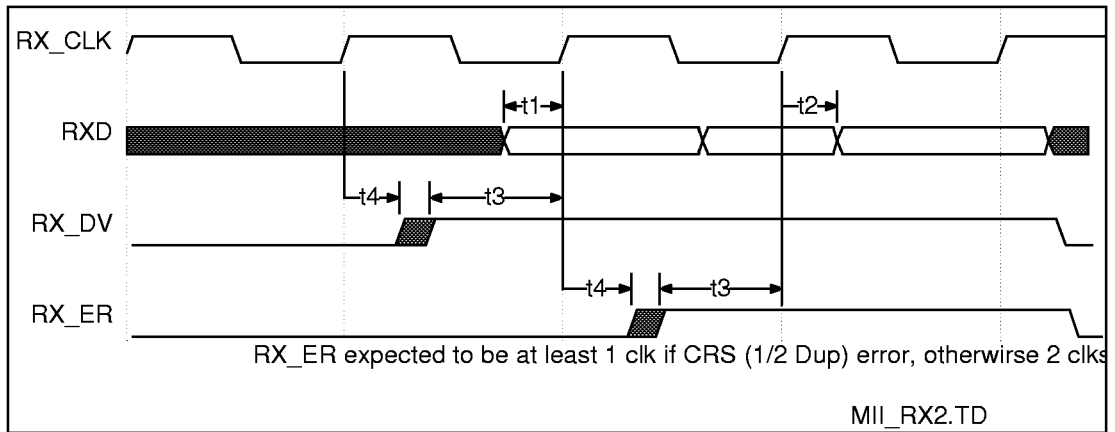
| NAME   | MIN                         | MAX                             | DESCRIPTION                           |
|--------|-----------------------------|---------------------------------|---------------------------------------|
| t1, t2 | 0ns                         | 25ns                            | Clock to output delay                 |
| t3     | 0ns(100Mb/s)<br>0ns(10Mb/s) | 40ns(100Mb/s)<br>400ns(10Mb/s)  | Min: 0 bit times<br>Max: 4 bit times  |
| t4     | 0ns(100Mb/s)<br>0ns(10Mb/s) | 160ns(100Mb/s)<br>1.6μs(10Mb/s) | Min: 0 bit times<br>Max: 16 bit times |



**FIGURE 18 - MII - RECEIVE TIMING FOR 100Mb/s**

Note: Clock Frequency Changes to 2.5MHz for 10Mb/s Nibble Transfers

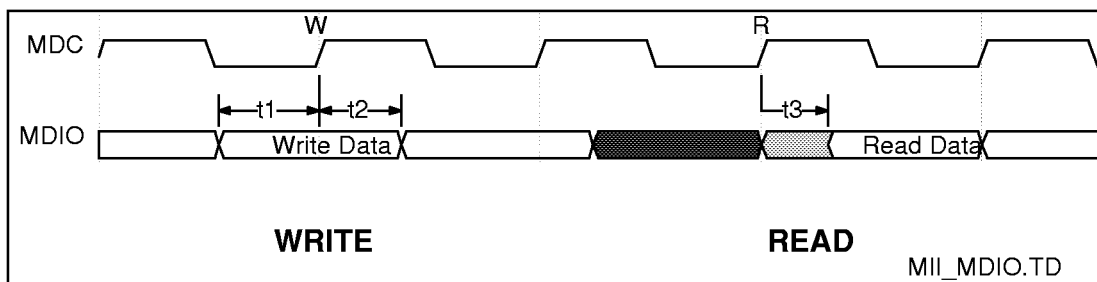
| NAME       | MIN  | MAX | DESCRIPTION      |
|------------|------|-----|------------------|
| t1, t3     | 10ns |     | Input Setup Time |
| t2, t4, t5 | 10ns |     | Input Hold Time  |



**FIGURE 19 - MII - RECEIVE ERROR (RX\_ER) TIMING FOR 100Mb/s**

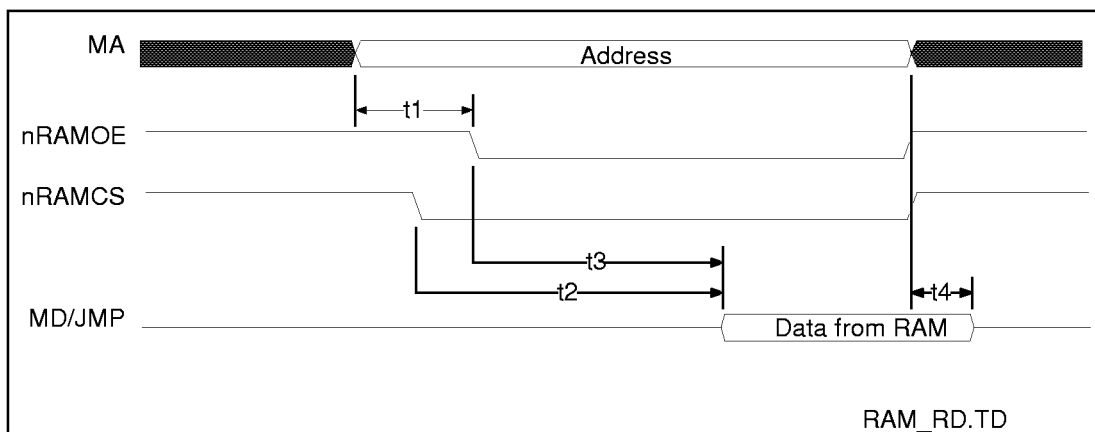
Note: Clock Frequency Changes to 2.5 MHz for 10 Mb/s Nibble Transfers

| NAME   | MIN  | MAX  | DESCRIPTION      |
|--------|------|------|------------------|
| t1, t3 | 10ns |      | Input Setup Time |
| t2, t4 |      | 10ns | Input Hold Time  |



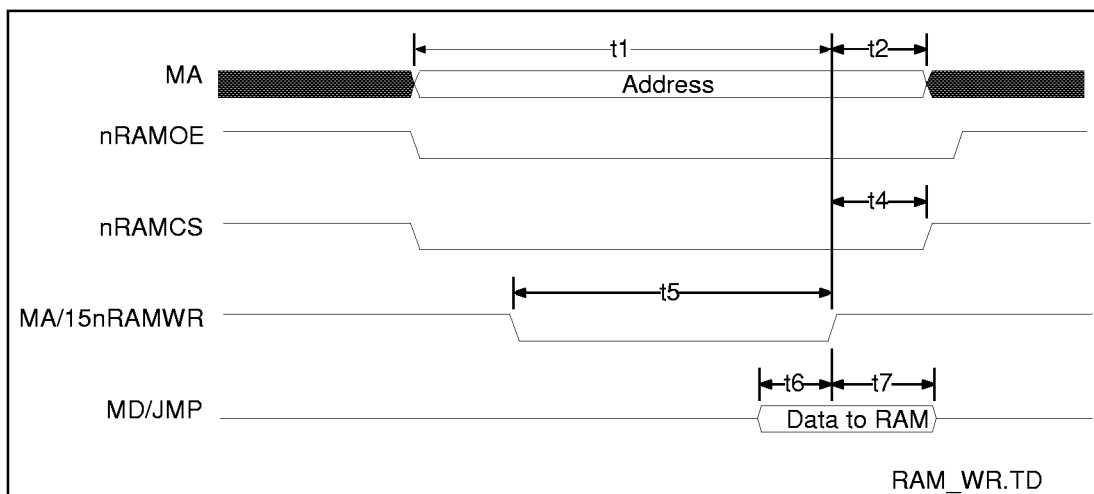
**FIGURE 20 - MII - SERIAL MANAGEMENT WRITE/READ**

| NAME | MIN  | MAX   | DESCRIPTION                                            |
|------|------|-------|--------------------------------------------------------|
| t1   | 10ns |       | Data ready before the rising edge of MCLK (Setup time) |
| t2   | 10ns |       | Data hold after the rising edge of MCLK                |
| t3   | 0ns  | 300ns | Data ready to MCLK (Required by EPIC)                  |



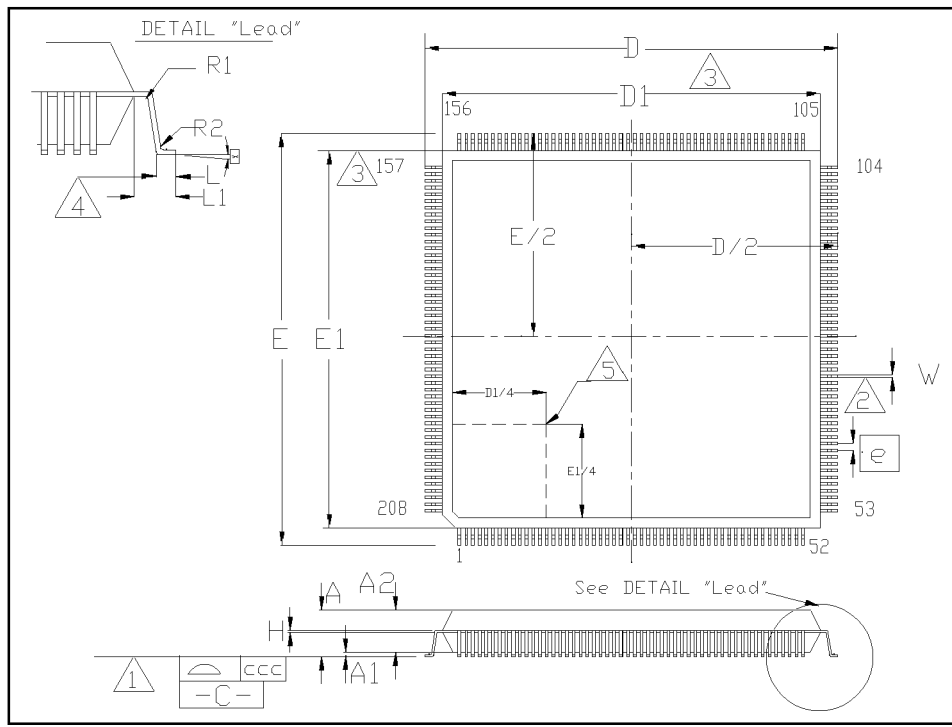
**FIGURE 21 - EXTERNAL(OPTIONAL) RAM READ**

| NAME | MIN | MAX   | DESCRIPTION                        |
|------|-----|-------|------------------------------------|
| t1   | 0ns |       | Address setup before output enable |
| t2   |     | 250ns | Chip select access time            |
| t3   |     | 250ns | Output enable to data valid        |
| t4   | 0ns |       | Output hold from address change    |



**FIGURE 22 - EXTERNAL (OPTIONAL) RAM WRITE**

| NAME | MIN   | MAX | DESCRIPTION                   |
|------|-------|-----|-------------------------------|
| t1   | 210ns |     | Address valid to end of write |
| t2   | 120ns |     | Address hold time             |
| t4   | 110ns |     | Chip select hold time         |
| t5   | 180ns |     | Write Pulse width             |
| t6   | 200ns |     | Data setup time               |
| t7   | 120ns |     | Data hold time                |



**FIGURE 23 - 208 PIN PQFP PACKAGE OUTLINE, 2.6 MM FOOTPRINT**

See Table on the following page.

## PACKAGE DIMENSIONS

|            | MIN        | NOMINAL      | MAX    | REMARK                                       |
|------------|------------|--------------|--------|----------------------------------------------|
| <b>A</b>   | ~          | ~            | 4.07   | Overall Package Height                       |
| <b>A1</b>  | 0.05       | ~            | 0.5    | Standoff                                     |
| <b>A2</b>  | 3.17       | ~            | 3.67   | Body Thickness                               |
| <b>D</b>   | 30.35      | 30.60        | 30.85  | X Span                                       |
| <b>D/2</b> | 15.175     | 15.30        | 15.425 | $\frac{1}{2}$ X Span Measure from Centerline |
| <b>D1</b>  | 27.90      | 28.00        | 28.10  | X Body Size                                  |
| <b>E</b>   | 30.35      | 30.60        | 30.85  | Y Span                                       |
| <b>E/2</b> | 15.175     | 15.30        | 15.425 | $\frac{1}{2}$ Y Span Measure from Centerline |
| <b>E1</b>  | 27.90      | 28.00        | 28.10  | Y Body Size                                  |
| <b>H</b>   | 0.90       | ~            | 0.23   | Lead Frame Thickness                         |
| <b>L</b>   | 0.35       | 0.50         | 0.65   | Lead Foot Length from Centerline             |
| <b>L1</b>  | ~          | 1.30         | ~      | Lead Length                                  |
| <b>e</b>   | 0.50 Basic |              |        | Lead Pitch                                   |
| <b>q</b>   | 0°         | ~            | 7°     | Lead Foot Angle                              |
| <b>W</b>   | 0.10       | ~            | 0.30   | Lead Width                                   |
| <b>R1</b>  | ~          | 0.20 or 0.15 | ~      | Lead Shoulder Radius                         |
| <b>R2</b>  | ~          | 0.30 or 0.20 | ~      | Lead Foot Radius                             |
| <b>ccc</b> | ~          | ~            | 0.08   | Coplanarity                                  |

### Notes:

- <sup>1</sup> Controlling Units: millimeter
- <sup>2</sup> Tolerance on the position of the leads is  $\pm 0.04$  mm maximum.
- <sup>3</sup> Package body dimensions D1 and E1 do not include the mold protrusion. Maximum mold protrusion is 0.25 mm.
- <sup>4</sup> Dimension for foot length L measured at the gauge plane 0.25 mm above the seating plane.
- <sup>5</sup> Details of pin 1 identifier are optional but must be located within the zone indicated.

1997© STANDARD MICROSYSTEMS  
CORP.

**smsc**<sup>™</sup>

Circuit diagrams utilizing SMSC products are included as a means of illustrating typical applications; consequently complete information sufficient for construction purposes is not necessarily given. The information has been carefully checked and is believed to be entirely reliable. However, no responsibility is assumed for inaccuracies. Furthermore, such information does not convey to the purchaser of the semiconductor devices described any licenses under the patent rights of SMSC or others. SMSC reserves the right to make changes at any time in order to improve design and supply the best product possible. SMSC products are not designed, intended, authorized or warranted for use in any life support or other application where product failure could cause or contribute to personal injury or severe property damage. Any and all such uses without prior written approval of an Officer of SMSC and further testing and/or modification will be fully at the risk of the customer.

LAN83C171 Rev. 08/26/97



### PACKAGE DIMENSIONS

|            | MIN        | NOMINAL      | MAX    | REMARK                                                     |
|------------|------------|--------------|--------|------------------------------------------------------------|
| <b>A</b>   | ~          | ~            | 4.07   | Overall Package Height                                     |
| <b>A1</b>  | 0.05       | ~            | 0.5    | Standoff                                                   |
| <b>A2</b>  | 3.17       | ~            | 3.67   | Body Thickness                                             |
| <b>D</b>   | 30.35      | 30.60        | 30.85  | X Span                                                     |
| <b>D/2</b> | 15.175     | 15.30        | 15.425 | <sup>1</sup> / <sub>2</sub> X Span Measure from Centerline |
| <b>D1</b>  | 27.90      | 28.00        | 28.10  | X Body Size                                                |
| <b>E</b>   | 30.35      | 30.60        | 30.85  | Y Span                                                     |
| <b>E/2</b> | 15.175     | 15.30        | 15.425 | <sup>1</sup> / <sub>2</sub> Y Span Measure from Centerline |
| <b>E1</b>  | 27.90      | 28.00        | 28.10  | Y Body Size                                                |
| <b>H</b>   | 0.90       | ~            | 0.23   | Lead Frame Thickness                                       |
| <b>L</b>   | 0.35       | 0.50         | 0.65   | Lead Foot Length from Centerline                           |
| <b>L1</b>  | ~          | 1.30         | ~      | Lead Length                                                |
| <b>e</b>   | 0.50 Basic |              |        | Lead Pitch                                                 |
| <b>q</b>   | 0°         | ~            | 7°     | Lead Foot Angle                                            |
| <b>W</b>   | 0.10       | ~            | 0.30   | Lead Width                                                 |
| <b>R1</b>  | ~          | 0.20 or 0.15 | ~      | Lead Shoulder Radius                                       |
| <b>R2</b>  | ~          | 0.30 or 0.20 | ~      | Lead Foot Radius                                           |
| <b>ccc</b> | ~          | ~            | 0.08   | Coplanarity                                                |

Notes:

<sup>1</sup> Controlling Units: millimeter

<sup>2</sup> Tolerance on the position of the leads is  $\pm 0.04$  mm maximum.

<sup>3</sup> Package body dimensions D1 and E1 do not include the mold protrusion. Maximum mold protrusion is 0.25 mm.

<sup>4</sup> Dimension for foot length L measured at the gauge plane 0.25 mm above the seating plane.

<sup>5</sup> Details of pin 1 identifier are optional but must be located within the zone indicated.



©1997 STANDARD MICROSYSTEMS  
CORP.



Circuit diagrams utilizing SMSC products are included as a means of illustrating typical applications; consequently complete information sufficient for construction purposes is not necessarily given. The information has been carefully checked and is believed to be entirely reliable. However, no responsibility is assumed for inaccuracies. Furthermore, such information does not convey to the purchaser of the semiconductor devices described any licenses under the patent rights of SMSC or others. SMSC reserves the right to make changes at any time in order to improve design and supply the best product possible. SMSC products are not designed, intended, authorized or warranted for use in any life support or other application where product failure could cause or contribute to personal injury or severe property damage. Any and all such uses without prior written approval of an Officer of SMSC and further testing and/or modification will be fully at the risk of the customer.

LAN83C171 Rev. 08/26/97