# 3C515-TX NIC Technical Reference

**Fast EtherLink® ISA 10/100BASE-TX network interface card with Parallel Tasking® technology**

**http://www.3com.com/**

Part No. 89-0764-000
Published March 1998

**3Com Corporation**
**5400 Bayfront Plaza**
**Santa Clara, California**
**95052-8145**



3Com Corporation reserves the right to revise this documentation and to make changes in content from time to time without obligation on the part of 3Com Corporation to provide notification of such revision or change.

3Com Corporation provides this documentation without warranty, term, or condition of any kind, either implied or expressed, including, but not limited to, the implied warranties, terms or conditions of merchantability, satisfactory quality, and fitness for a particular purpose. 3Com may make improvements or changes in the product(s) and/or the program(s) described in this documentation at any time.

If there is any software on removable media described in this documentation, it is furnished under a license agreement included with the product as a separate document, in the hard copy documentation, or on the removable media in a directory file named LICENSE.TXT or !LICENSE.TXT. If you are unable to locate a copy, please contact 3Com and a copy will be provided to you.

**UNITED STATES GOVERNMENT LEGEND**

*If you are a United States government agency, then this documentation and the software described herein are provided to you subject to the following:*

All technical data and computer software are commercial in nature and developed solely at private expense. Software is delivered as "Commercial Computer Software" as defined in DFARS 252.227-7014 (June 1995) or as a "commercial item" as defined in FAR 2.101(a) and as such is provided with only such rights as are provided in 3Com's standard commercial license for the Software. Technical data is provided with limited rights only as provided in DFAR 252.227-7015 (Nov 1995) or FAR 52.227-14 (June 1987), whichever is applicable. You agree not to remove or deface any portion of any legend provided on any licensed program or documentation contained in, or delivered to you in conjunction with, this User Guide.

Unless otherwise indicated, 3Com registered trademarks are registered in the United States and may or may not be registered in other countries.

3Com, EtherLink, and Parallel Tasking are registered trademarks of 3Com Corporation.

Atmel is a registered trademark of Atmel Corporation.

All other company and product names may be trademarks of the respective companies with which they are associated.

# CONTENTS

## 5 REGISTERS

# FIGURES

# TABLES

# 1 INTRODUCTION

The *3C515-TX NIC Technical Reference* is a supplement to the latest version of the following 3Com® technical reference:

*PCI/EISA Bus-Master Adapter Driver Technical Reference*
*EtherLink® III and Fast EtherLink NICs*
Part Number 09-0681-001B, published May 1995

Together, these technical references define the programming interface for the 3Com 3C515-TX NIC. See Table 1-1.

**Table 1-1** 3Com 3C515-TX NIC

| NIC Number | Description | Cable Specification | Connector Type |
|---|---|---|---|
| 3C515-TX | Fast EtherLink ISA 10/100BASE-TX network interface card with Parallel Tasking® technology | 10BASE-T: UTP Category 3, 4, or 5<br>100BASE-TX: UTP Category 5 | One RJ-45 |

## Features

The 3C515-TX NIC has the following features:

- Parallel Tasking technology
- 10 or 100 Mbps autonegotiation (NWay)
- Support for automatic driver installation, depending on the operating system
- Status LEDs
- One shielded RJ-45 connector for both 10 and 100 Mbps speeds
- Network management support
- ISA plug-and-play (PnP) compatibility

## About This Technical Reference

This technical reference contains information that software engineers, independent software developers, and test engineers can use when writing device drivers, diagnostic programs, and production test software.

## Terms and Acronyms

The following terms and acronyms are used in 3Com technical references:

| | |
|---|---|
| BIST | Built-in self test. |
| Byte | An 8-bit wide quantity of data. |
| Double word (dword) | A 32-bit wide quantity of data (4 bytes). |
| Download | The process of transferring transmit data from system memory to the NIC. |
| DPD | Download packet descriptor. |
| FLP | Fast link pulse. |
| FSH | Frame start header. |
| Indications | The reporting of any interesting event on the NIC. Any indication may be configured to cause an interrupt. |
| Interrupts | The actual assertion of the host machine's interrupt signal. |
| MAC | Media Access Control. |
| MII | Media-independent Interface. |
| NIC | Network interface card. |
| PEROM | Programmable and erasable read-only memory. |
| PHY | IEEE designation for Physical layer. |
| UPD | Upload packet descriptor. |
| Upload | The process of transferring receive data from the NIC to system memory. |
| Word | A 16-bit wide quantity of data (2 bytes). |

## Register Bit Maps

The register descriptions in this technical reference include register bit maps.

As illustrated in Figure 1-1, unshaded areas in a register bit map indicate bits that disregard data written to them and return zeros when read. To ensure compatibility with future hardware, drivers should write zeros to these bits.

Shaded areas indicate active register bits.

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | | | | | | | | | | | | | |

**Figure 1-1** Sample Register Bit Map

Numeric values are presented in the following format:

| Format | Description | Examples |
|---|---|---|
| #'$_r$ZZZZ | # is the number of bits | 6'$_b$100101 is a 6-bit binary notation. |
| | ' is a delimiter | $30_d$ is a decimal notation. |
| | r is the radix (b for binary, d for decimal, and h for hexadecimal) | 6'$_h$25 is a 6-bit hexadecimal notation. |
| | ZZZZ is the value | |

# 2 ARCHITECTURE

This chapter describes the 3C515-TX NIC architecture and summarizes the layout of the registers and windows.

## Block Diagram

The 3C515-TX NIC contains two ASICs:

- Main controller—controls overall NIC operations
- ISA bridge ASIC—a PCI-to-ISA bridge chip that controls ISA bus operations

The 3C515-TX NIC is illustrated in the block diagram in Figure 2-1.

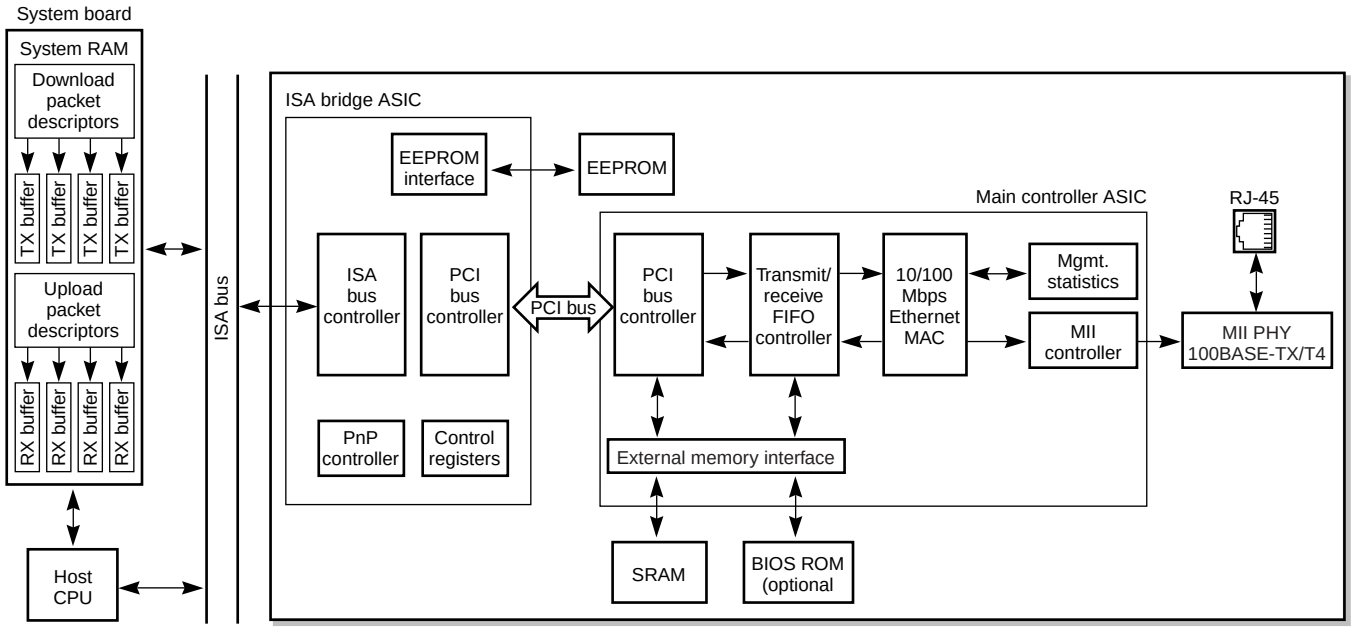


**Figure 2-1** 3C515-TX NIC Architecture

The NIC components are described in the following sections.

## Main Controller ASIC

This section describes the various functions within the main controller ASIC. See the block diagram in Figure 2-2.

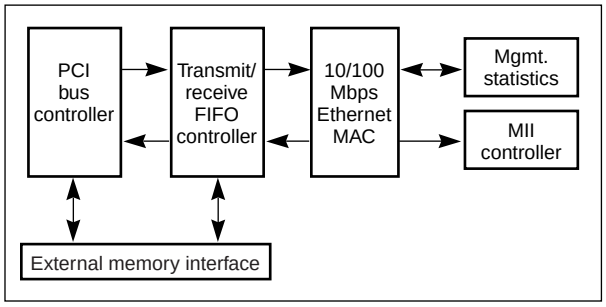


**Figure 2-2** Main Controller ASIC Block Diagram

**PCI Bus Controller** This block implements the PCI interface functions (responding to PCI target cycles, generating and controlling PCI master cycles, and performing parity checking and generation).

**Transmit/Receive FIFO Controller** This block controls the buffer space, which is partitioned for both transmission and reception.

**10/100 Mbps Ethernet MAC** This block implements the IEEE 802.3 Media Access Control (MAC) function. It is responsible for the media access protocol, including deference, collision recovery and back off, receive packet filtering, and error detection. This block also provides information to the management statistics function.

**Management Statistics** This block accumulates various network events statistics in hardware. Driver software reads these statistics periodically to maintain a network management information base.

**MII Controller** This block contains the logic required to support the Media Independent Interface portion of the IEEE 802.3u standard. This allows the ASIC to connect to any physical signaling chip that supports the MII standard.

**External Memory Interface** This block controls access to external memory, which provides up to 128 KB of buffer space. The NIC is shipped with 64 KB of buffer space.

## ISA Bridge ASIC

This section describes the various functions within the ISA bridge ASIC. Refer to the block diagram in Figure 2-3.

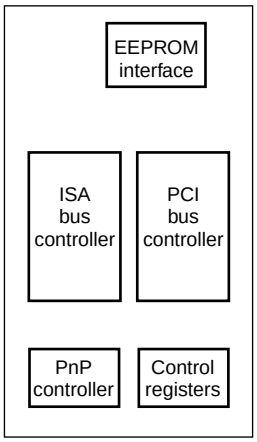


**Figure 2-3** ISA Bridge ASIC Block Diagram

**EEPROM Interface** This block allows access to the EEPROM.

**ISA Bus Controller** This block implements the ISA interface functions.

**PCI Bus Controller** This block allows access to the PCI bus.

**PnP Controller** This block controls plug-and-play functions.

**Control Registers** This block controls the functions of the ISA bridge ASIC.

## Other NIC Devices

The other NIC devices are described below.

**SRAM** This block contains up to 128 KB of packet buffering RAM.

**BIOS ROM** The optional BIOS ROM can contain up to 64 KB of code that is executed at system boot time.

NICs usually have a BIOS ROM socket that allows field installation and upgrade of the boot code. Support for Atmel PEROMs in the BIOS ROM socket allows boot code to be updated without opening up the host computer.

## Registers and Windows

Figure 2-4 shows a register map for the 3C515-TX NIC. A register's location is specified by its offset from a base address that is defined in the IoBaseAddress PCI register in the main controller ASIC. Table 2-1 shows the 3C515-TX NIC ISA bridge ASIC register layout. For details on the main controller registers and window registers, see *PCI/EISA Bus-Master Adapter Driver Technical Reference* (part number 09-0681-001B, published May 1995).

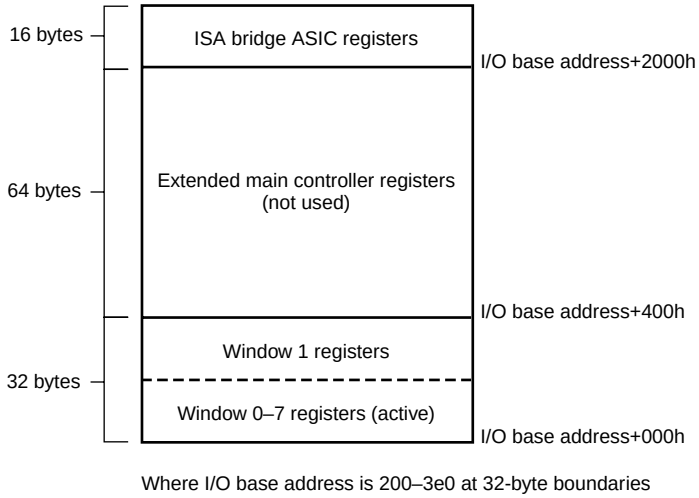


**Figure 2-4** 3C515-TX Register Map

**Table 2-1** 3C515-TX NIC ISA Bridge ASIC Register Layout

| Byte 1 | Byte 0 | Offset |
|---|---|---|
| | | e |
| EEPROM Data | | c |
| EEPROM Command | | a |
| | Debug | 8 |
| | MCR | 6 |
| | ROM Page | 4 |
| RCR | | 2 |
| DCR | | 0 |

# 3 OPERATION

This chapter describes ISA bus operational characteristics.

## System Reset

When a GlobalReset command or Host Reset command is sent to the main controller ASIC, the bus master and PCI state machines are reset in the ISA bridge chip. A GlobalReset command sent to the ISA bridge chip forces an initialization and requires the PnP reset sequence. The default PnP resource data values appear in Table 3-1.

**Table 3-1** Default PnP Resource Data Values

| Location | Value | Location | Value |
|---|---|---|---|
| 1e | 1782 | 2a | 4153 |
| 1f | 3300 | 2b | 5015 |
| 20 | 6f43 | 2c | 506d |
| 21 | 206d | 2d | 0251 |
| 22 | 6146 | 2e | a822 |
| 23 | 7473 | 2f | 2a9e |
| 24 | 4520 | 30 | 06e8 |
| 25 | 6874 | 31 | 0047 |
| 26 | 7265 | 32 | 0200 |
| 27 | 694c | 33 | 03e0 |
| 28 | 6b6e | 34 | 2020 |
| 29 | 4920 | 35 | c079 |

After a reset to the ISA bridge chip, the window in the main controller ASIC is undefined, and the software must reinitialize this window.

## Forced Configuration Mode

The 3C515-TX NIC must be initialized for the first time in forced configuration mode so that the serial EEPROM can be programmed with the default values for the I/O base address, DMA channel request line (DRQ), interrupt request line (IRQ), and PnP. Thereafter, the NIC can be initialized in normal mode. In forced configuration mode, the diagnostic software or the driver writes the DRQ and IRQ values to the ISA bridge chip registers. The characteristics of forced configuration mode are:

- The FCFGN pin is strapped to GND.
- The I/O base address defaults to 200h for main controller I/O accesses.
- The NIC is enabled after the automatic initialization sequence is finished.
- The main controller's PCI configuration space is written.
- RomSize information is not loaded from EEPROM or written to the main controller.

## Normal Mode

The characteristics of normal mode are:

- The I/O base address, DRQ, and IRQ default information is loaded from EEPROM into the ISA bridge chip registers.
- The main controller's PCI configuration space is written and initialized.
- The diagnostic software or the driver runs the PnP sequence to enable NIC access.

# 4 EEPROM

Table 4-1 lists the EEPROM contents.

**Table 4-1** 3C515-TX EEPROM Contents

| Offset (hex) | Description | Default |
|---|---|---|
| 00 | 3Com Node Address (word 0) | 0020 |
| 01 | 3Com Node Address (word 1) | afxx |
| 02 | 3Com Node Address (word 2) | xxxx |
| 03 | Resource Config Reg | 010a |
| 04 | Manufacturing Data - Date | xxxx |
| 05 | Manufacturing Data - Division | 00xx |
| 06 | Manufacturing Data - Product Code | xxxx |
| 07 | Manufacturer ID | 6d50 |
| 08 | DMA Config Reg | 0447 |
| 09 | Device ID | 5960 |
| 0a | OEM Node Address Word 0 | 0020 |
| 0b | OEM Node Address Word 1 | afxx |
| 0c | OEM Node Address Word 0 | xxxx |
| 0d | Software Info | 3f10 |
| 0e | Compatibility Word | 0000 |
| 0f | Software Info 2 | 0000 |
| 10 | Capabilities Word | 11c6 |
| 11 | Reserved | 0000 |
| 12 | Internal Config Word 0 | 001b |
| 13 | Internal Config Word 1 | 0161 |
| 14 | Reserved | 0000 |
| 15 | MII Info | 0003 |
| 16 | Reserved | 0000 |
| 17 | Checksum | 00yy |
| 18 | PnP - Vendor ID | 6d50 |
| 19 | Product ID | 5960 |
| 1a | Serial ID 0 (Node Address 2) | xxxx |
| 1b | Serial ID 1 (Node Address 1) | xxxx |
| 1c | PnP Version - Checksum | 0axx |
| 1d | PnP Version - ID String (3Com Fast EtherLink ISA) | 1010 |
| | Logical Device ID | 0294 506d 5015 |
| | Compatible Device ID | f7 80d0 411c |
| | IRQ Descriptor | 9e a822 |
| | DRQ Descriptor | 06 e82a |

# 5 REGISTERS

This chapter provides definitions of the 3C515-TX registers associated with ISA operation.

## DCR

| | |
|---|---|
| Synopsis | Configures DMA bus cycles. This register is loaded from EEPROM after a system reset. |
| Type | Read/write |
| Size | 16 bits |
| Offset | 0 |

**DCR Register Format**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | | | | | | | | | | | | | | | |

**DCR Bit Descriptions**

| Bit | Name | Description |
|---|---|---|
| [14:13] | chRdyWait | Specifies the number of clock cycles, in 40-nanosecond (ns) increments, to wait after the IOCHRDY ISA bus signal is active before terminating the current bus cycle. This delay is in addition to any delay incurred when the IOCHRDY signal is synchronized as it comes from the ISA bus. The increment values are:<br><br>00 = 40 ns<br>01 = 80 ns<br>10 = 120 ns<br>11 = 160 ns |
| [12:11] | recoveryWait | Specifies the period that the ISA MEMR/MEMW command is inactive between cycles. The values are:<br><br>00 = 80 ns<br>01 = 120 ns<br>10 = 160 ns<br>11 = 200 ns |
| [10:8] | cmdWidth | Specifies the period that the ISA MEMR/MEMW command is active. Valid values are 000 through 111 in 40 ns increments (000 = 40 ns, 111 = 320 ns). |
| [7:3] | burstLength | Stores the number of dword transfers per bus master cycle. Valid values are 5'h1f through 5'h02. |
| [2:0] | drqSelect | Specifies selection of the DMA channel request. The values are:<br><br>$\{3,5,6,7\}_{10}$ = Selects DRQs<br>$\{0,1,2,4\}_{10}$ = Disables DRQs |

## Debug

| | |
|---|---|
| Synopsis | Provides for debugging of bus errors. |
| Type | Read |
| Size | 8 bits |
| Location | I/O base address+2008 |
| Offset | 8 |

**Debug Register Format**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | | |

**Debug Bit Descriptions**

| Bit | Name | Description |
|---|---|---|
| [1] | isaBusFault | Indicates an ISA bus error. |
| [0] | pciBusFault | Indicates a PCI bus error. |

## EepromCommand

| | |
|---|---|
| Synopsis | Allows commands to be issued to the serial EEPROM controller. |
| Type | Read/write |
| Size | 16 bits |
| Offset | a |

**EepromCommand Register Format**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | | 0 | | | | | | | | |

**EepromCommand Bit Descriptions**

| Bit | Name | Description |
|---|---|---|
| [9] | eeBusy | Indicates EEPROM status. This is a read-only bit. The values are:<br>0 = EEPROM is not busy (off).<br>1 = EPROM is busy (on; I/O writes to the EepromCommand register are disabled).<br>For IC test purposes, this bit serves as the speedUpClk312K bit.<br>The eeBusy bit must be off (0) before a command can be written to the EepromCommand register or data can be written to the EepromData register. |
| [7:0] | eeCommand | Shifts commands to the on-board EEPROM. See Table 5-1. |

**Table 5-1** eeCommand Commands

| Command | Op Code (2 bits) | Address (6 bits) | Data |
|---|---|---|---|
| Read Register | 10 | a(5:0) | Yes |
| Write Register | 01 | a(5:0) | Yes |
| Erase Register | 11 | a(5:0) | No |
| Erase/Write Enable | 00 | 11xxxx | No |
| Erase/Write Disable | 00 | 00xxxx | No |
| Erase All Registers | 00 | 10xxxx | No |
| Write All Registers | 00 | 01xxxx | Yes |

For all Erase and Write commands in Table 5-1, the hardware times the 10-ns write strobe and then automatically executes the Erase/Write Disable command. The Erase/Write Enable command provides protection from accidental writes to the EEPROM. Software must wait for the eeBusy bit to go off (0) before writing the next command.

The Erase commands write only ones into the EEPROM; the Write commands write only zeros. To write data into an EEPROM word, follow this sequence:

**1** Issue an Erase/Write Enable command.

**2** Issue an Erase command.

**3** Issue an Erase/Write Enable command.

**4** Load the data into the EEPROMdata register.

**5** Issue a Write command.

## EepromData

| | |
|---|---|
| Synopsis | Provides data access for the EEPROM. |
| Type | Read/write |
| Size | 16 bits |
| Offset | c |

Write data must be written to the EepromData register before the Write command is given to the EEPROM. Read data can be read to EepromData after the eeBusy bit in the EepromCommand register goes off.

## MCR

| | |
|---|---|
| Synopsis | Provides master control. |
| Type | Read/write |
| Size | 8 bits |
| Offset | 6 |

**MCR Register Format**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | | | |

**MCR Bit Descriptions**

| Bit | Name | Description |
|---|---|---|
| [2] | readPrefetch | Enables lookahead bus master reads. |
| [1] | writeBuffer | Enables buffered bus master writes. |
| [0] | pciConfig | Enables access to PCI configuration. When this bit is set, software I/O accesses to the main controller are converted to pciConfig space access cycles. |

## RCR

| | |
|---|---|
| Synopsis | Configures ISA bus resources. Loaded from EEPROM after system reset. |
| Type | Read/write (except bits 14 and 15, which are read-only) |
| Size | 16 bits |
| Offset | 2 |

**RCR Register Format**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 0 | | | | | | | | | | | | | |

**RCR Bit Descriptions**

| Bit | Name | Description |
|---|---|---|
| [15:14] | romSize | Specifies the size of the BIOS ROM on the NIC. Loaded from the InternalConfig register in EEPROM after a system reset. BIOS ROM sizes are:<br>0 = 8 KB<br>1 = Reserved<br>2 = 32 KB<br>3 = 64 KB |
| [12:9] | romBase | Determines the I/O base address of the BIOS ROM, as shown in Table 5-2. |

(continued)

**RCR Bit Descriptions** (continued)

| Bit | Name | Description |
|---|---|---|
| [8:5] | ioBase | Selects the I/O base address for the NIC. The I/O base address can be calculated as follows:<br>Bit 4 must be programmed as a 0 to force the ioBase bit to be on even boundaries.<br>*ioBase* 0 through $30_d$ =<br>I/O base address = *iobase[8:5]* $\times$ $20_h$ + $200_h$<br>*ioBase* $31_d$ = Reserved |
| [3:0] | irqSelect | Selects the interrupt, as follows:<br>$\{3,5,7,9,10,11,12,15\}_{10}$ = Selects IRQs<br>$\{0,1,2,4,6,8,13,14\}_{10}$ = Disables IRQs |

**Table 5-2** BIOS ROM Base Addresses

| ROM Size | ROM Base | ROM Window |
|---|---|---|
| $00_b$ (8K) | $0000_b$ | Boot ROM disabled |
| | $0001_b$ | $C2000_h$–$C3FFF_h$ |
| | $0010_b$ | $C4000_h$–$C53FFF_h$ |
| | $0011_b$ | $C6000_h$–$C7FFF_h$ |
| | $0100_b$ | $C8000_h$–$C9FFF_h$ |
| | $0101_b$ | $CA000_h$–$CBFFF_h$ |
| | $0110_b$ | $CC000_h$–$CDFFF_h$ |
| | $0111_b$ | $CE000_h$–$CFFFF_h$ |
| | $1000_b$ | $D0000_h$–$D1FFF_h$ |
| | $1001_b$ | $D2000_h$–$D3FFF_h$ |
| | $1010_b$ | $D4000_h$–$D5FFF_h$ |
| | $1011_b$ | $D6000_h$–$D7FFF_h$ |
| | $1100_b$ | $D8000_h$–$D9FFF_h$ |
| | $1101_b$ | $DA000_h$–$DBFFF_h$ |
| | $1110_b$ | $DC000_h$–$DDFFF_h$ |
| | $1111_b$ | $DE000_h$–$DFFFF_h$ |
| $11_b$ (64K) | $0000_b$ | Boot ROM disabled |
| | $0001_b$ | $C0000_h$–$C3FFF_h$ |
| | $001x_b$ | $C4000_h$–$C7FFF_h$ |
| | $010x_b$ | $C8000_h$–$CBFFF_h$ |
| | $011x_b$ | $CC000_h$–$CFFFF_h$ |
| | $100x_b$ | $D0000_h$–$C3FFF_h$ |
| | $101x_b$ | $D4000_h$–$D7FFF_h$ |
| | $110x_b$ | $D8000_h$–$DBFFF_h$ |
| | $111x_b$ | $DC000_h$–$DFFFF_h$ |
| $10_b$ (32K) | See coding for 64K | |

## RomPage

| | |
|---|---|
| Synopsis | Controls ROM page information. |
| Type | Read/write |
| Size | 8 bits |
| Offset | 4 |

**RomPage Register Format**

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | | | |

**RomPage Bit Descriptions**

| Bit | Name | Description |
|---|---|---|
| [2] | pewrEnable | Controls write access to the PEROM.<br>1 = Enable write accesses.<br>0 = Disable write accesses. |
| [1:0] | romPage | Controls which 16K ROM page is visible to the host through the ROM space in upper memory. |

# INDEX OF REGISTERS AND COMMANDS

# INDEX OF BITS