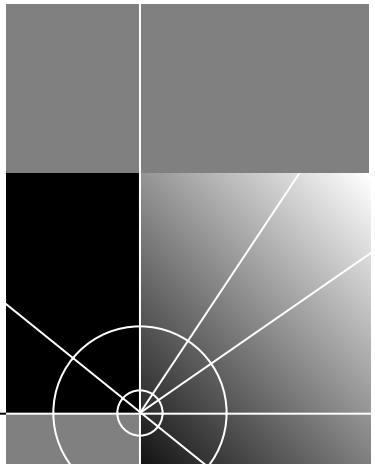
EtherLink® III LAN PC Card ERS

Members of the 3Com EtherLink III family of adapters



http://www.3com.com/

Published May 1997

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 of any kind, either implied or expressed, including, but not limited to, the implied warranties of merchantability 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.

UNITED STATES GOVERNMENT LEGENDS:
If you are a United States government agency, then this documentation and the software described herein are provided to you subject to the following restricted rights:

For units of the Department of Defense:
Restricted Rights Legend: Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) for Restricted Rights in Technical Data and Computer Software Clause at 48 C.F.R. 52.227-7013. 3Com Corporation, 5400 Bayfront Plaza, Santa Clara, California 95052-8145.

For civilian agencies:
Restricted Rights Legend: Use, reproduction, or disclosure is subject to restrictions set forth in subparagraph (a) through (d) of the Commercial Computer Software – Restricted Rights Clause at 48 C.F.R. 52.227-19 and the limitations set forth in 3Com Corporation's standard commercial agreement for the software. Unpublished rights reserved under the copyright laws of the United States.

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. If you are unable to locate a copy, please contact 3Com and a copy will be provided to you.

[Portions of this documentation are reproduced in whole or part with permission from (as appropriate).]

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

3Com, AccessBuilder, Boundary Routing, LANplex, LanScanner, LinkBuilder, NETBuilder, NETBuilder II, Parallel Tasking, ViewBuilder, EtherDisk, EtherLink, EtherLink Plus, EtherLink II, SmartAgent, TokenLink, TokenLink Plus, TokenDisk and Transcend are registered trademarks of 3Com Corporation. 3TECH, CacheCard, FDDILink, FMS, NetProbe and Star-Tek are trademarks of 3Com Corporation. 3ComFacts is a service mark of 3Com Corporation

CompuServe is a registered trademark of CompuServe, Inc. OpenView is a registered trademark of Hewlett-Packard Co. AIX, IBM, and NetView are registered trademarks of International Business Machines Corporation. UNIX is a registered trademark of Novell Inc. OpenWindows, SunNet Manager, and SunOS are trademarks of Sun MicroSystems Inc. SPARCstation is a trademark and is licensed exclusively to Sun Microsystems Inc.

Other brand and product names may be registered trademarks or trademarks of their respective holders.

CONTENTS

5 WINDOW SET

6 REGISTER DEFINITIONS

FIGURES

TABLES

1 INTRODUCTION

Overview

This document describes the basic architecture of the 3Com® EtherLink® III Parallel Tasking® 3C589 LAN PC Card. Table 1-1 depicts the bus type, cable specifications, and connector types.

Table 1-1 3C589 PC Card

Bus Type	PC Card Number	Connectors per PC Card	Cable Specification	Connector Type
PCMCIA	3C589D-TP	One connector	10BASE-T	RJ-45
	3C589D-COMBO	Two connectors	10BASE-T Thin coax	RJ-45 BNC

This document is designed for software engineers, independent software developers, and test engineers. Aspects of the architecture have patents applied for.

This PC Card is part of the EtherLink III family of high-performance adapters. The "D" in the 3C589D PC Card number indicates that this PC Card is the fifth generation of the EtherLink III LAN PC Card known as the 3C589 PC Card.

Features of the PC Card include:

- 3Com-designed Ethernet controller, encoder/decoder, 10BASE-T transceiver, and host interface integrated into one ASIC
- 4 K packet buffer
- High performance in client/server applications
- 8 K packet buffer
- VCO (voltage-controlled oscillator)
- 8-bit and 16-bit PCMCIA bus path
- Reduced power consumption
- Power Management (added for 3C589D release)
  - Conservative Power Management

    The PC Card wakes up on any packet received regardless of whether it is valid or not, and will not go back to sleep until 6.4us after any packets.
  - Aggressive Power Management

    The clock stays on as long as the PC Card is ready to transmit. If the software set the PMmode bit, the clock will be turned off for the duration of a rejected packet, even if the chip is ready to transmit.

3C589D PC Card Differences

This section summarizes the changes between the 3C589C and 3C589D PC Cards.

Internal 8K RAM

The 8K SRAM is inside the ASIC. The CIS will now be inside the chip after Power On Reset.

Package Reduction

Since there are no external address, data, and control signals for external RAM, chip pins are reduced from 128 to 100 pins. With the 100 pin package, chip size is reduced. Below is a list of the eliminated pins for the 3C589D PC Card:

- LA[15:0]
- LD[7:0]
- A[19:17, 15:13]
- ROMCS
- RAMCS,
- MEMOEB
- MEMEN
- 5 N/C pins

This is a list of the pins added for the 3C589D PC Card:

- MO_WAIT
- MO_STSCHG
- MO_IOR
- MO_IOWR
- MO_WE
- MO_OE_RDY
- MO_RDY
- MO_CLK
- MO_RESET
- MO_HALT

Added Multifunction Support

LAN and MODEM multifunction support is added. Modem registers are available on chip.

Added PCMCIA Power Down Bit

PMmode (power management mode bit) bit is added to Internal Configuration Register(Window 3/Port 00h); power up or reset to 0, meaning conservative power management; software can set to 1 for more aggressive power management.

Removed PnP

Since PCMCIA does not use PnP; this logic is removed to reduce required real estate.

Register Modified

The following are modified registers for 3C589D PC Card:

- Internal Configuration Register(Window 3/Port 00h)
- Address Configuration (Window 0/Port 06h)
- RomControl Register (Window 3/Port 05h)
- Configuration Control (Window 0/Port 04h)
- Resource Configuration (Window 0/Port 08h)
- PCMCIA COR
- PCMCIA CSR

Registers Added

The following are new registers added for 3C589D PC Card:

- AltEepromCommandRegister (Window 2/Port 0Ah)
- LanIOBASE0
- LanIOBASE1
- ModemCOR
- ModemCSR
- ModemIOBASE0
- ModemIOBASE1
- ModemLIMIT
- ModemPRR
- ModemESR
- Multi-function Configuration Register(Window 2/Port 08h)
- SRAM Diagnostic Register(Window 2/Port 06h)

Specification References

For detailed information on Plug and Play and PCMCIA standards, refer to the appropriate source listed below:

- Plug and Play ISA Specification Version 1.0: Microsoft Corporation, One Microsoft Way, Redmond, WA 98052-6399, Tel: (800) 426-9400.
- PCMCIA: Personal Computer Memory Card International Association, 1030 East Duane Avenue, Suite G, Sunnyvale, CA 94086, Tel: (408) 720-0107, Fax: (408) 720-9416.

Refer to Appendix A for a series of process flowcharts that will provide useful information for software developers.

2 ARCHITECTURAL OVERVIEW

The 3C589 PC Card is designed to be an efficient, low-latency network PC Card optimized for server and client environments. This PC Card has 8 K of buffer space on board, which appears as two dedicated FIFOs to the host (one for transmit and one for receive). The two dedicated FIFOs are of variable size. Various early indication/early start mechanisms are incorporated to improve performance and make operation possible while minimizing the likelihood of overruns or underruns.

The network transceiver consists of a dedicated receiver and a dedicated transmitter, allowing loopback operation at full network bandwidth. At the 10BASE-T interface, automatic polarity reversal and hardware link beat LED indications are supported. At the register level, support is provided for critical network management functions as well as supplementary 10BASE-T functions.

To support a variety of platforms and operating systems, programmed I/O (PIO) is the only method of data transfer supported.

Interrupts can be programmed to signal the CPU under various early indication conditions, and timer mechanisms are incorporated to allow measurement of system latencies. These features reduce latency and optimize performance in platforms where this is appropriate.

A high-level command interface provides for detailed management at the PC Card while using only 16 bytes of I/O space.

The PCMCIA Card Information Structure (CIS) provides information about how the PC Card may be configured. Actual configuration of the card is performed by a slot controller in the host system. This configuration process is usually managed by the card driver communicating with the Card Services software layer supplied on the host. A functional configuration is provided by Card Services each time the card is configured.

In addition, the driver is required to set EtherLink III registers on the PCMCIA card to enable the card. The basis for these register values is provided in the EEPROM on the card.

For detailed information on PCMCIA, refer to the PCMCIA specification, which is available from the Personal Computer Memory Card International Association. Refer to Chapter 1 for address, telephone, and fax information.

Block Diagrams

Refer to Figures 2-1 - 2-2 for block diagrams of the 3C589 PC Cards.

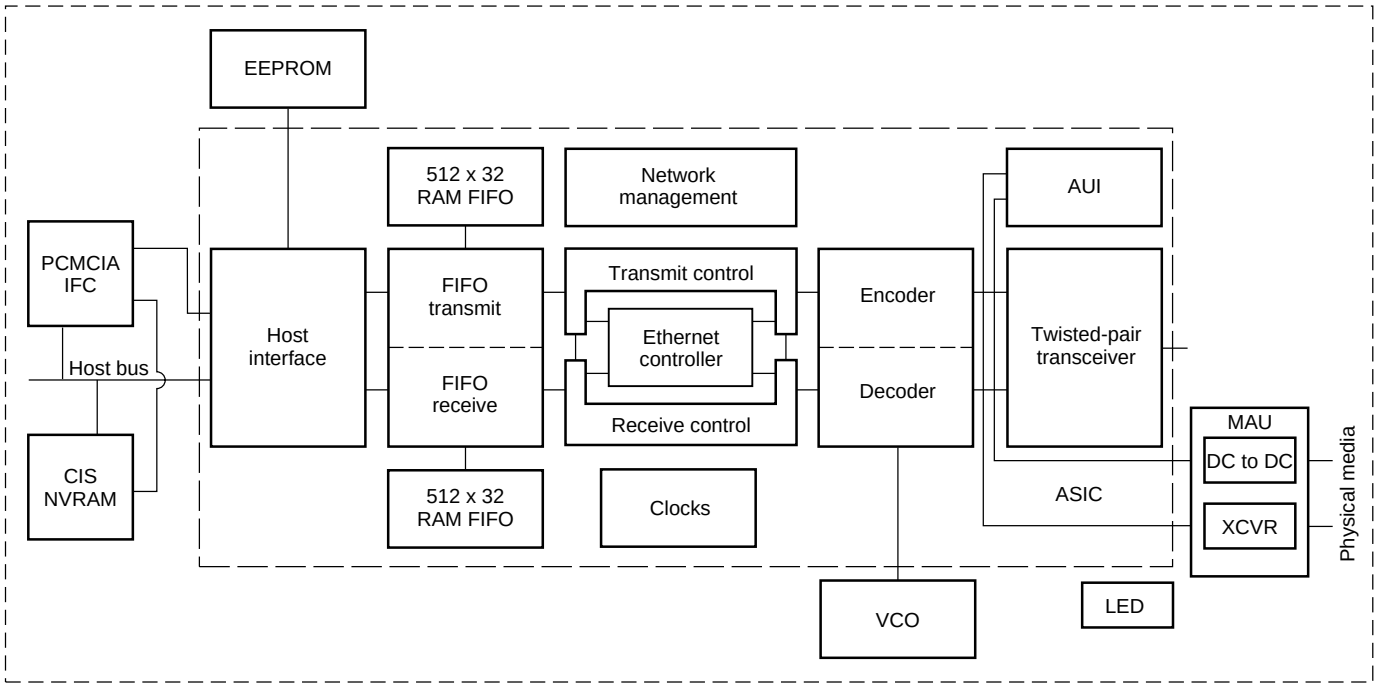


Figure 2-1 Block Diagram for 3C589

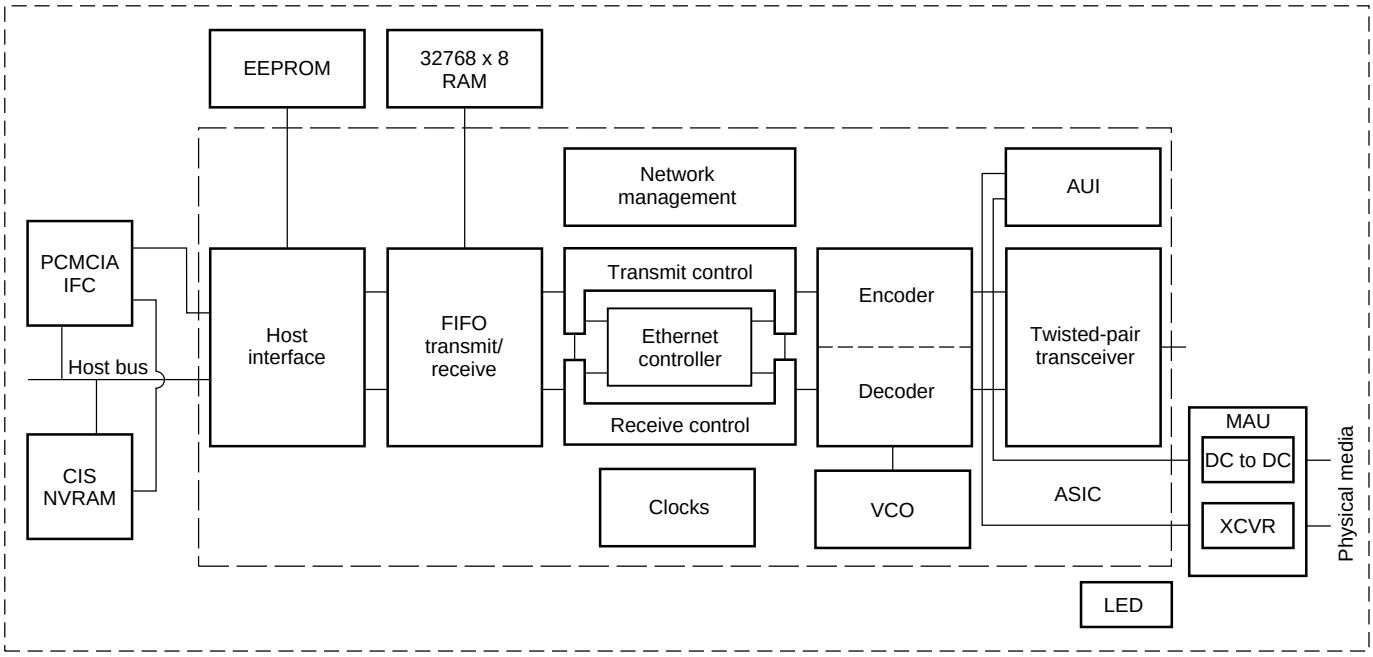


Figure 2-2 Block Diagram for 3C589B

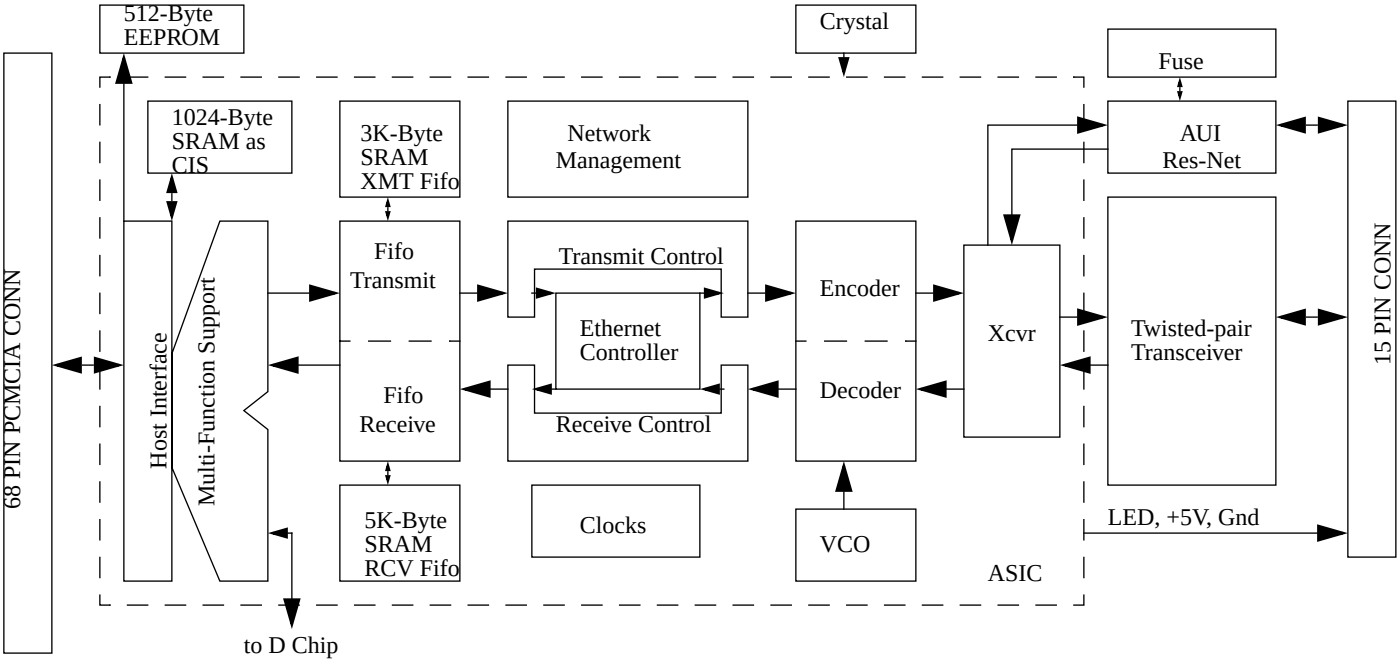


Figure 2-3 Block Diagram for 3C589D

Attribute Memory Map

The highest address bit of the PC Card (A11) does not need to be connected to the A11 of the PC-Card bus. In particular, if it is connected to A16 of the PC-Card bus, the attribute registers will appear at 0x10000 in attribute space, enabling compatibility with previous 3C589 drivers.

3 DATA STRUCTURES

This chapter describes the data structures used for the PC Card's FIFO.

The PC Card's SRAM is organized into two FIFOs, a TX FIFO for transmit and an RX FIFO for receive. The size of these FIFOs depends on the size of the SRAM used and how space has been allocated between the transmit and receive FIFOs.

Packets being received are written to the RX FIFO by the PC Card, and read out of the RX FIFO by the host software. Packets to be transmitted are written to the TX FIFO by the host software and read out of the TX FIFO by the PC Card.

Multiple packets can be contained within either FIFO, depending on the size of the packets and the storage available.

Packets begin on, and are padded to, double word boundaries. The amount of unused space is visible in the free byte registers: TX Free and RX Free. The next two sections describe the packet structure for receive and transmit packets.

Receive Packet Structure

Figure 3-1 shows a model of the receive packet structure for PCMCIA (3C589) as it is stored in the FIFO. The RX Status register can be used to determine the number of bytes of data in the packet.

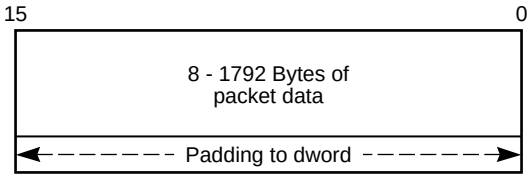


Figure 3-1 Receive Packet Structure

- Packet data is contained in the first 8 to 1,792 bytes (packets smaller than 60 bytes or larger than 1,514 bytes are flagged as receive errors: runts for the former, oversize for the latter).
- Padding to a double word (dword) boundary follows. These bytes can be read by the host if desired.
- There may be additional bytes per packet, which are hardware overhead. These should be ignored by the host.

Transmit Packet Structure

Figure 3-2 shows a model of the transmit packet structure for PCMCIA (3C589) as it is stored in the FIFO.

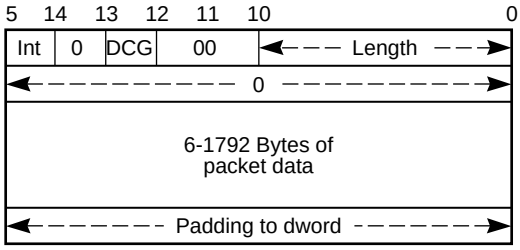


Figure 3-2 Transmit Packet Structure

Preceding the packet data is the transmit preamble, which is two words long. The first word contains the length of the packet in bytes, the Interrupt on Successful TX Completion bit (bit 15), and the Disable CRC Generation bit. The Interrupt on Successful TX Completion bit indicates whether the host wants an interrupt when the packet has been successfully transmitted (it will get one even if there is an error). This bit is maskable in the Interrupt Mask register. The Disable CRC Generation bit specifies whether the packet data following contains the 4-byte CRC or not. If not (the default), it will be generated automatically by the hardware. This bit may be used by bridging software to avoid regenerating the CRC. Bits that are currently unassigned must be written as zeros.

- The second word of the transmit preamble will be implemented as "don't cares." For future compatibility it is recommended that software write the second word as zeros.
- Packet data follows in the next 6 to 1,792 bytes (packets smaller than 14 bytes are not legal Ethernet packets; those bigger than 1,514 are oversized).
- Padding to a double word boundary is last.
- The packet length is in bytes. If it is not a multiple of 4, pad bytes must be added to bring the packet data in the TX FIFO to a 4-byte boundary. The packet length, however, reflects the true size of the packet in bytes.

Packets that are less than 60 bytes in length will be padded automatically by the hardware.

Example: A 14-byte packet would require a preamble and 16 bytes. Automatic padding will produce a 60-byte packet for transport onto the wire.

4 DESCRIPTION OF OPERATION

The data transfer mode for the 3C589D PC Card is programmed I/O (PIO). As data is received off the wire, it accumulates in the receive FIFO. The driver reads the data off the PC Card a byte, word, or double word at a time through the PIO Data Read register. Similarly, transmit data can be written to the PC Card a byte, word, or double word at a time, and it accumulates in the transmit (TX) FIFO. Once a packet has been transmitted out of the TX FIFO, it is discarded and the space it occupied is available.

The transmit and receive algorithms are described below.

Receive

Receive Packet and RX Status

Each packet in the RX FIFO consists of the packet data, padded to a dword boundary. If the host reads beyond the end of the packet, the RX Bytes field of the RX Status register becomes negative.

The RX Status register always maintains the status of the packet at the head of the FIFO.

The driver transfers all of the receive data from the FIFO to host memory via PIO.

After using PIO to read the data, the host must issue an RX Discard command to update the RX Status register for the next packet.

The driver can take interrupts when RX Early threshold bytes of a packet have been received, or when an entire packet has been received.

The driver can determine the size of the packet (number of bytes received so far) at the head of the RX FIFO by reading the RX Status register. Once the packet has been completely received, any errors that occurred during reception are posted in the RX Status register. The packet size can only be determined from RX Status when the packet is "complete."

Receive Early

The RX Early interrupt will be enabled only if the protocol interface allows for early receive indications. However, the driver can take the RX Early interrupt in order to compute the packet size (generally only for 802.3 packets) and generate the equivalent of an early RX Complete indication.

The driver may choose to set the RX Early threshold to slightly less than the protocol's early lookahead size to overlap the reception of the final bytes with the interrupt latency. On entry to the interrupt handler, the RX Status register can be examined to see whether the threshold needs to be adjusted.

Once RX Early has been set, the driver has only to acknowledge the interrupt and it will not become active again (unless reprogrammed higher) until RX Status is updated for the next packet.

The driver must transfer the data from the RX FIFO via PIO into a dedicated lookahead area to make the data addressable to the protocol stack.

Receive Complete

If an RX Early indication is given for the packet, then the lookahead bytes have already been read in. Otherwise, they must be read in now, using PIO to transfer them to the dedicated lookahead area to make the data addressable to the protocol stack.

Once a scatter descriptor is available from the protocol, the data must be copied out of the lookahead area and the RX FIFO.

Transmit

PIO is the only mode of operation supported for transmit. This mode is discussed in more detail below.

Transmit Packet and Transmit Completion

Each packet in the TX FIFO consists of a transmit preamble followed by the transmit data padded to a dword boundary.

The transmit preamble consists of two words. The first word specifies the length of the packet in bytes, whether or not to generate an interrupt when the packet is transmitted successfully, and (for 3C589B) whether the CRC bytes are to be generated by the hardware or are contained within the packet. The second word is reserved, and should be set to zero.

The packet length is the actual number of bytes in the FIFO to be sent to the wire, excluding any padding to dwords. If the packet is less than the minimum length (60 bytes not including CRC), it need not be padded by the software. Instead, the PC Card will pad the packet to the minimum length before giving it to the Ethernet controller.

The driver will typically request an interrupt on successful transmit completion only if the protocol has requested a confirmation. The driver must queue up such requests to retrieve the protocol identification for the transmit confirmation. When any Transmit Complete interrupt comes in, the driver examines the TX Status register. This completion is for the head of the queue if the Interrupt Requested bit is set in TX Status, since transmits complete in order. The driver must serialize this process to guarantee that it works properly.

If an error occurs while the packet is being transmitted, the PC Card always generates an interrupt and disables the transmitter. If the host determines from TX Status that the confirmation is not for a queued request, it can use the error information to update its statistics counters. In any case, the host must manually restart the transmitter by issuing the TX Enable command once it has emptied the TX Status stack as the result of an error.

Transmitting a Packet

All data to be transmitted must be moved into the FIFO by the driver. Multiple packets can be moved into the FIFO as long as the host is prepared to deal with running out of TX FIFO space.

Typically the driver will copy as much of the packet to the PC Card as possible. If one or more packets precede this one in the FIFO, then the FIFO can run out of space. If this happens, the driver can issue the TX Available command to request an interrupt once there is sufficient space for the rest of the packet.

The Set TX Available Threshold command causes the PC Card to generate an interrupt when the specified number of bytes is available in the TX FIFO. This allows the driver to return and continue copying the data later, when some of the data in the TX FIFO has been transmitted.

When TX Available is used, the possibility of an underrun always exists if the interrupt latency is high enough (for example, 1 ms on switching windows under OS/2®). If underruns are a problem, they can be avoided by reprogramming the TX Start threshold to be greater than the number of bytes transferred to that point. That way the driver can guarantee that the packet will not start transmitting before the TX Available interrupt is serviced. The driver can be written so that it makes this adjustment only when the amount of the packet copied to the FIFO is small enough for concern (that is, less than the number of bytes that can be transmitted within the measured interrupt latency), or the driver can make the adjustment semipermanently (reset on some timer tick multiple) whenever an underrun occurs.

When a packet is currently being copied to the FIFO or if the FIFO is full, and the protocol issues another transmit request, the driver will have to queue the gather descriptor. However, this queue—the awaiting download queue—is different from the transmit completion queue described above. The awaiting download queue is emptied and the queue entries released by copying the data to the PC Card.

Disable CRC Generation (3C589B)

A typical driver will leave the DCG bit reset so the Ethernet controller will automatically generate the CRC for the packet that is to be transmitted. The DCG bit should only be set if the host software generates its own CRC, or if the host software is performing a bridging function, which must forward packets that include the original CRC.

When this bit is set, packets must be padded manually to a dword boundary by the driver. Also, packets must be 64 bytes in length, including CRC.

Transmit Underrun

When the TX FIFO underruns, the PC Card will generate a bad CRC for the packet. A Transmit Complete interrupt will be generated to the driver, specifying a transmit underrun error. When the driver detects this error, it must first issue a TX Reset command before using TX Enable to reenable the transmitter.

Whenever the driver encounters an underrun, it should take special care to guarantee that the next transmit packet does not underrun by setting the TX Start threshold large enough so that it does not start until the packet is completely copied to the PC Card. For subsequent packets, the driver may react quickly to changes in the operating environment by adjusting the TX Start threshold according to new information on interrupt latencies, and other system performance measurements. The exact mechanism used is beyond the scope of this document.

Underruns occur only when the packet is being copied to the PC Card. Therefore, the driver should be able to retransmit the packet. Checking the number of underruns and retransmissions can greatly reduce the performance impact of an underrun (an occasional underrun is acceptable).

5 WINDOW SET

The 3C589 PC Card register set consists of several 8-word register windows. The windows are numbered, and each window presents a different register set to the host through the standard 16-byte I/O space of the adapter.

At power-up, or after a Global Reset, Window 0 is the working register set. Window 0 contains setup information, including the registers reflecting the EEPROM setup information. Window 1 contains the standard register set, which includes all of the registers used on the driver critical path.

Accessing other windows will require switching away and then back again. The driver will switch to Window 1 during initialization and assume that this set is always current from that point on. Other windows contain the statistics information, report the adapter state, allow for reconfiguration, and support various diagnostics.

A soft reset from the host will also select Window 0 as the working register set.

Window 0 Registers – Setup

This window contains configuration registers, including setup and EEPROM access.

	Write function		Read function	
Port offset	High byte	Low byte	High byte	Low byte
0E	Command		Window (0-7)	Status
0C	EEPROM data		EEPROM data	
0A	EEPROM command		EEPROM command	
08	Resource configuration*†		Resource configuration	
06	Address configuration*†		Address configuration	
04	Configuration control*		Configuration control	
02			Product ID*†/Adapter ID§	
00			Manufacturer ID	

* ISA/EISA PCMCIA adapter

† PCMCIA adapter. The values are not loaded automatically at power-up and must be explicitly initialized by the driver software. See Chapter 7.

§ MCA adapter

Figure 5-1 Setup

Changes for 3C589D

- Modified Configuration Control (Window 0/Port 04h)
- Modified Address Configuration (Window 0/Port 06h)
- Modified Resource Configuration (Window 0/Port 08h)

Window 1 Registers – Operating Set

This window set is assumed to be on critical path. The TX Status and Timer registers must be read separately as byte registers. The TX PIO and RX PIO registers may be written and read as bytes, words, or double words.

	Write function		Read function	
Port offset	High byte	Low byte	High byte	Low byte
0E	Command		Window (0-7)	Status
0C			Free transmit bytes	
0A	TX Status		TX Status	Timer
08			RX Status	
06				
04				
02	TX PIO data write		RX PIO data read	
00	TX PIO data write		RX PIO data read	

Figure 5-2 Operating Set

Window 2 Registers – Station Address Setup/Read

The station address must be read out of the EEPROM and written into these registers before reception is enabled.

	Write function		Read function	
Port offset	High byte	Low byte	High byte	Low byte
0E	Command		Window (0-7)	Status
0C				
0A	AltEEprom Command Reg		AltEEprom Command Reg	
08	Multi-function Configuration Reg		Multi-function Configuration Reg	
06	SRAM Diagnostic		SRAM Diagnostic	
04	Address 5	Address 4	Address 5	Address 4
02	Address 3	Address 2	Address 3	Address 2
00	Address 1	Address 0	Address 1	Address 0

Figure 5-3 Station Address Setup/Read

Changes for 3C589D

- Added SRAM Diagnostic Register (Window 2/Port 06h)
- Added Multi-function Configuration Register (Window 2/Port 08h)
- Added AltEEpromCommand Register (Window 2/Port 0Ah)

Window 3 Registers – FIFO Management

Write function (3C509, 3C579, 3C529, 3C589)

Port offset	High byte	Low byte
0E	Command	
0C		
0A		
08		
06		
04		
02		
00		

Read function (3C509, 3C579, 3C529, 3C589)

High byte	Low byte
Window (0-7)	Status
Free transmit bytes	
Free receive bytes	
Resource configuration	
Tx reclaim threshold*	

*MCA adapter

Read function (3C509B, 3C589B)

Port offset	High byte	Low byte
0E	Command	
0C		
0A		
08		
06		
04	ROM control*	
02	Internal configuration	
00		

*3C509B only

Read function (3C509B, 3C589B)

High byte	Low byte
Window (0-7)	Status
Free receive bytes	
ROM cotnrol*	
Internal configuration	

*3C509B only

Figure 5-4 FIFO Management

Changes for 3C589D

- Modified Internal Configuration Register (Window 3/Port 00h)
- Modified RomControl Register (Window 3/Port 05h) (R/W)

Window 4 Registers – Diagnostic

Write function

Port offset	High byte	Low byte
0E	Command	
0C		
0A	Media type and status	
08	Ethernet controller status	
06	Net diagnostic	
04	FIFO diagnostic	
02		
00		

Read function

High byte	Low byte
Window (0-7)	Status
Media type and status	
Ethernet controller status	
FIFO diagnostic	

Figure 5-5 Diagnostic

Window 5 Registers – Command Results and Internal State

This window contains registers that allow the parameters set by command to be read back for diagnostic purposes. These registers can be accessed as bytes as well as words.

Port offset	Write function (All) High byte	Write function (All) Low byte	Read function (3C509, 3C579, 3C589) High byte	Read function (3C509, 3C579, 3C589) Low byte
0E	Command		Window (0-7)	Status
0C			Read zero mask	
0A			Interrupt mask	
08			RX filter (lower 4 bits only)	
06			RX early threshold	
04				
02			TX available threshold	
00			TX starts threshold + 4	

Port offset	Read function (3C509B, 3C529, 3C589B) High byte	Read function (3C509B, 3C529, 3C589B) Low byte
0E	Window (0-7)	Status
0C	Read zero mask	
0A	Interrupt mask	
08	RX filter (lower 4 bits only)	
06	RX early threshold	
04		
02	TX available threshold	
00	TX starts threshold + 4	

Figure 5-6 Command Results and Internal State

Window 6 Registers – Statistics

Reading a statistic also zeroes it. These registers may be read only while statistics collection has been disabled temporarily. Statistics that are word-sized must be read as words, and those that are bytes must be read as bytes. Writing to these registers is supported for debugging purposes. These registers can be accessed as bytes as well as words. Refer to the section "Statistics Registers" in Chapter 6 for more information.

	Write function		Read function	
Port offset	High byte	Low byte	High byte	Low byte
0E	Command		Window (0-7)	Status
0C	Total bytes transmitted OK		Total bytes transmitted OK	
0A	Total bytes received OK		Total bytes received OK	
08		Transmit deferrals		Transmit deferrals
06	Frames received OK	Frames transmitted OK	Frames received OK	Frames transmitted OK
04	Received overruns	Late collision on transmit	Received overruns	Late collision on transmit
02	Frames transmitted after exactly one collision	Frames transmitted after multiple collisions	Frames transmitted after exactly one collision	Frames transmitted after multiple collisions
00	Frames transmitted no CD heartbeat (SQE)	Carrier sense lost during transmission	Frames transmitted no CD heartbeat (SQE)	Carrier sense lost during transmission

Figure 5-7 Statistics Maintained by the Adapter

Added Registers for 3C589D

- LAN IOBASE 0 (0x80a)
- LAN IOBASE 1 (0x80c)
- ModemCOR (0x900)
- ModemCSR (0x902)
- Modem PRR (0x904)
- Modem ESR (0x908)
- Modem IOBASE 0 (0x90a)
- Modem IOBASE 1 (0x90c)
- Modem IOLIMIT (0x912)
- Multi-function Configuration Register

6 REGISTER DEFINITIONS

This chapter describes the function and use of each register needed in normal operation of the 3C589 PC Card. The adapter configuration and enable registers in Window 0 are explained in Chapter 7.

PCMCIA (3C589, 3C589B)

The registers defined in this section are those of the EtherLink III ASIC portion of the PCMCIA board. PCMCIA-specific registers are defined in Chapter 7.

Command Register

Function: Issues commands to PC Card.

Location: All windows/Port 0E

Type: Write only

Size: 16 bits

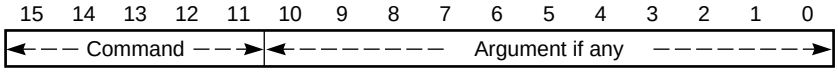


Figure 6-1 Command Register

Bit Description:

Bits 15–11	5-bit code for command to be executed.
Bits 10–0	11-bit argument if any. For commands with no arguments, these bits should be set to zero for future compatibility.

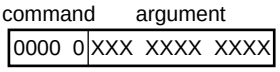
Command Register Conventions

The Command register controls many PC Card functions. For example, a command is used to switch windows. In general, commands must be written as a word. However, the following is acceptable for commands with no parameters:

```
mov   dx,   PortCmdStatus
mov   ah,   CMD_X
out   dx,   ax
```

Commands, except those marked with an asterisk (*), execute in one I/O cycle. For commands that are not completed in one cycle, the software must poll the Command-in-Progress bit in the Status register to determine when one of these commands is completed. Since only a single command can be outstanding at any given time, the host must do this in a critical section.

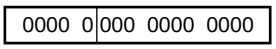
The commands are listed below. The 16 bits of each command register are given after each command name, in this format:



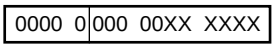
The 5-bit command code is specified in binary. The 11-bit argument follows the vertical bar in the format aaa aaaa aaaa. Also:

- An x signifies a variable bit in the argument.
- A 0 signifies a bit that must be zero.
- A z signifies either a 0 or a 1, but will be treated as a 0.
- A 1 signifies either a 0 or a 1, but will be treated as a 1.

Global Reset*



(3C589)



(3C589B)

PCMCIA (3C589)

This command has the same effect as a power-up reset. It is implemented with a minimum amount of logic to ensure success. This command requires a significant amount of time to execute, since it involves rereading the EEPROM. The Timer register may not be used for this delay period and the Command-in-Progress bit will not be visible. The host must wait at least 1 ms after issuing this command before it touches the PC Card again. The argument to this command must be all zeros.

(3C589B)

This command argument masks the Global Reset command to a module if the mask bit is set. The command has the same effect as a power-up reset for the modules which are reset and has no effect on modules which are masked.

The Command-in-Progress bit will not be visible if a Host Reset command is executed. If the argument is all zeros, this Global Reset command has the same effect as the PCMCIA Global Reset command, and it resets all of the modules.

The command argument is defined as follows:

Bit 0: TPAUI Reset. Resets the 10BASE-T and AUI transceivers.

Bit 1: EnDec Reset. Resets the internal encoder/decoder.

Bit 2: Network Reset. Resets the Network module, including the Ethernet controller.

Bit 3: FIFO Reset. Resets the FIFO module.

Bit 4: AISM Reset. Resets the Auto Initialize state machine. This forces the EEPROM to be reread unless the PC Card is in test mode.

Bit 5: Host Reset. Resets the bus interface, excluding the Auto Initialize state machine. It does not reset the POS registers, so the current base addresses remain valid.

In general, host software should reset all modules, using a command argument of zero (0). The other arguments are typically intended for debugging.

PCMCIA (3C589, 3C589B)

The Global Reset command resets the EtherLink III ASIC but not the PCMCIA interface chip. All ASIC registers must be reloaded following this command.

Select Register Window

0000 1	000 0000 0XXX

This command selects register window xxx. The current register window is available in the Status register. After power-up or Global Reset, which includes host reset, Window 0 is in effect.

Start Coaxial Transceiver

0001 0	000 0000 0000

This command affects only the 10BASE2 operation. It starts the DC-DC converter that drives the on-board coaxial Ethernet transceiver. After power-up, the coaxial transceiver must be started manually with this command. The host must delay at least 800 µs after issuing this command before using the coaxial transceiver. This can be accomplished by starting the timer and waiting until it pegs at its maximum value of FFh. The host must read the Address Configuration register (refer to the "Address Configuration Register" section in Chapter 7) from the EEPROM to determine whether to issue this command.

PCMCIA (3C589, 3C589B)

The Start Coaxial Transceiver command has only one function in the PCMCIA card: it turns on the LED.

RX Disable

0001 1	000 0000 0000

This command disables the Ethernet controller receiver. If a packet is in the process of being received, it will be received and the Ethernet controller receiver will be disabled after the packet has been completely received. To enable the receiver, use RX Enable. After power-up, the receiver is in the disabled state.

RX Enable

0010 0	000 0000 0000

This command enables the Ethernet controller receiver. If a packet is in the process of being transmitted on the wire, it will not be received. To disable the receiver, use RX Disable or RX Reset. After power-up, the receiver is in the disabled state and must be enabled with this command.

RX Reset

0010 1	000 0000 0000

(All)

0010 1	000 0000 XXXX

(3C589B)

Do not issue RX Reset unless it is absolutely required.

All

This command empties the RX FIFO, disables the Ethernet controller, resets the RX Filter and RX Early threshold to defaults, and aborts reception if a packet is currently being received. The argument to this command must be all zeros.

PCMCIA* (3C589B)

This command argument masks the RX Reset command to a module if the mask bit is set. The argument is defined as follows:

Bit 0: TPAUI RX Reset. Resets the 10BASE-T and AUI transceiver logic.

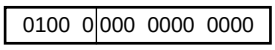
Bit 1: EnDec RX Reset. Resets the encoder/decoder receive logic.

Bit 2: Network RX Reset. Resets the Network Receive logic. Aborts any current packet reception. This includes disabling the Ethernet controller and receiver. Resets RX Filter to disable.

Bit 3: FIFO RX Reset. Resets the FIFO Receive logic. This includes emptying the RX FIFO and resetting RX Filter to disable and RX Early Threshold to default.

Set mask bits to zeros for compatibility with all EtherLink III adapters.

RX Discard Top Packet*



This command discards the remainder of the top packet in the RX FIFO. The host can also use this command to discard packets that are in error or are not needed. One reason for discarding packets is that they do not match any multicast address currently enabled through the protocol interface.

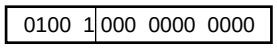
If the packet has not been completely received, the PC Card will ignore the remainder of the packet as if the receiver had been disabled, then reenabled.

If the packet has been completely received, use the RX Status register to update any statistics before issuing this command.

The driver must issue this command at the completion of each packet received regardless of whether the packet reception was in error.

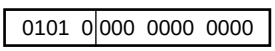
The RX Discard command pops the current RX Status and replaces it with the next packet status, if any. It is used to get from one packet to the next in the RX FIFO.

TX Enable



This command enables the Ethernet controller transmitter. It does not initiate transmission of a packet, which will not occur until at least the TX Start threshold bytes of the packet (or the entire packet) are in the TX FIFO. At power-up, the transmitter is disabled and must be enabled with this command. To disable the transmitter, use TX Disable. Transmit errors will also disable the transmitter.

TX Disable



This command disables the Ethernet controller transmitter. If a packet is currently being transmitted (that is, it has been presented to the Ethernet controller), issuing this command will not stop the transmission. The transmitter will be disabled after the packet has been completely transmitted (or has reached a nonrecoverable error). The transmitter is disabled on power-up, and can also become disabled as a result of a transmit error.

TX Reset*

0101 1	000 0000	0000	(All)
0101 1	000 0000	XXXX	(3C589B)
0101 1	X00 0000	XXXX	(3C589B)

This command, when bits 0–3 are all zeros, empties the TX FIFO, disables the Ethernet controller transmitter, and resets the TX Available and TX Start thresholds to default values. If a packet is currently being transmitted, the transmit will be aborted.

It is recommended that an argument of all zeros be used to ensure compatibility with all EtherLink III PC Cards.

After an underrun or jabber error on transmit, a TX Reset is required. Do not issue TX Reset unless it is absolutely required.

PCMCIA (3C589B)

This command resets the specified transmit logic when any of bits 0–3 are nonzero.

Below are the bit positions of the various reset masks in the TX Reset command field with the internal signal name and a brief description. When any of these bits are set, the reset to the corresponding transmit modules is masked (that is, that module will not be reset).

The argument is defined as follows:

Bit 0: TP AUI TX Reset. Resets the 10BASE-T and AUI transmit logic.

Bit 1: EnDec TX Reset. Resets the encoder/decoder transmit logic.

Bit 2: **Network TX Reset. Resets the network transmit logic. This includes the TX Status stack, and also disables the Ethernet controller and transmitter (equivalent to TX Disable).**

3C589B only: If a packet is in the midst of being transmitted, Network TX Reset will be asserted according to the value of bit 10. If bit 10 = 0, the reset will be delayed until the transmission has been completed (Command in Progress will be asserted during this period), or is about to be retried. If FIFO Reset is also asserted, the TX FIFO will be emptied immediately, causing any partially transmitted packet to be sent out with a guaranteed bad CRC (no error will be visible in TX Status, since that is reset by Network TX Reset). In this case Command in Progress will not be asserted for more than 6 µs or so. If bit 10 = 1, the transmit will be aborted immediately without guaranteeing a bad CRC, though a good CRC is highly unlikely.

Bit 3: FIFO TX Reset. Resets the FIFO transmit logic. This includes emptying the TX FIFO and resetting TX Available and TX Start thresholds to defaults.

PCMCIA (3C589B)

TX Reset is required after a Power Up command is issued while the chip is in a Power Down Full state.

If a packet is being transmitted when this command is issued, the reset will occur immediately or be delayed, depending on the value of bit 10:

Bit 10: 0 = Transmit abort is delayed until a guaranteed bad CRC can be generated on the packet.

1 = Transmit will be aborted immediately. A bad CRC is not guaranteed to be generated. See discussion above.

Request Interrupt

0110 0	000 0000 0000

This command sets the Interrupt Requested bit in the Status register, causing an interrupt whenever that interrupt bit is unmasked.

Acknowledge Interrupt

0110 1	000 XXXX XXXX

This command acknowledges the interrupt reasons specified in the argument. These bits are laid out identically to those in the Status register. If a bit is set in the argument, it acknowledges the reason for that interrupt, and will in some cases turn off the bit in the Status register. In other cases, the bit in the Status register is wired to the PC Card state, and that state must be changed in order to turn the bit off. In this case, setting the bit in the argument has no effect. Multiple bits may be set in the argument. If a bit is set in the argument and that bit is not set in the Status register, nothing happens. For each bit, the following specifies whether or not acknowledging it will force the Status bit off.

- Interrupt Latch | 0110 1 | 000 0000 0001 |

  This command turns off the Status register bit, releasing the interrupt. This bit must be acknowledged after all the interrupt reasons have been processed or masked off by setting the Interrupt Mask to zero (see Set Interrupt Mask).

- Adapter Failure | 0110 1 | 000 0000 0010 |

  This command does nothing. The bit indicating the reason for the adapter failure in the FIFO Diagnostic register (bit 13 or bit 10) must be cleared to recover from this state.

- TX Complete | 0110 1 | 000 0000 0100 |

  This command does nothing. The host must write the TX Status register (popping it) to turn the bit off (assuming there are no other transmit completions pending).

- TX Available | 0110 1 | 000 0000 1000 |

  This command turns off the TX Available bit and resets the TX Available threshold to its disabled value. The Set TX Available Threshold command must be reissued each time it is required.

- RX Complete | 0110 1 | 000 0010 0000 |

  This command does nothing. To turn off the Status register bit, the host should read the packet out of the RX FIFO and must issue an RX Discard command.

- RX Early | 0110 1 | 000 0010 0000 |

  This command turns off the Status register bit. RX Early will remain off for the duration of this packet unless the RX Early threshold is reprogrammed. To change the RX Early threshold without having this bit turn on again, reprogram the threshold before acknowledging the RX Early bit. See the Set RX Early Threshold command for more details.

- Interrupt Requested | 0110 1 | 000 0100 0000 |

  This command turns off the Status register bit.

- Update Statistics | 0110 1 | 000 1000 0000 |

  This command does nothing. To turn off the Status register bit, read the statistics. This will reset them all to zero.

Set Interrupt Mask

| 0111 0 | 000 XXXX XXXZ |

This command sets the Interrupt mask; each bit that is set enables interrupts from that interrupt source. To mask off all interrupts from the PC Card, set the Interrupt mask to zero. When an interrupt reason is masked off, the corresponding interrupt bit in the Status register is still readable, although it is no longer a source for interrupts. Use the Set Read Zero mask command to force the bits to read as zero. The bits are laid out identically to their locations in the Status register. At power-up, the Interrupt mask defaults to zero.

Set Read Zero Mask

| 0111 1 | 000 XXXX XXXZ |

This command sets the Read Zero mask; each bit that is clear causes the corresponding bit in the Status register to read as zero. The Read Zero mask is applied to the Status register before the Interrupt mask. Clearing a bit in the Read Zero mask also prevents it from causing interrupts.

To force all interrupt sources to zero, set the Read Zero mask to zero. Use the Set Interrupt Mask command to disable interrupts and still allow the bit in the Status register to be readable. The bits are laid out identically to their locations in the Status register. At power-up, the Read Zero mask defaults to zero.

The Interrupt Latch bit cannot be forced to zero with this command.

Set RX Filter

1000 0	000 0000 XXXX

This command sets the Receive filter as follows:

0001 Individual address

0010 Group (multicast) addresses

0100 Broadcast address

1000 All addresses (promiscuous mode)

At power-up, the Receive filter defaults to zero and must be set with this command before any packets can be received. Enabling group address reception implies broadcast reception. There is no individual filtering of group/multicast addresses. Promiscuous mode implies all others.

Set RX Early Threshold

1000 1	XXX XXXX XXZZ

This command sets a value for the RX Early threshold. When the number of bytes in the RX FIFO exceeds the argument (whether occurring before or after this command is issued), an RX Early interrupt will be generated to the host. A multiple of four with a range of 0 to 1,792 bytes is used. To disable, set to any value greater than 1,792. The PC Card truncates the argument to a dword multiple.

The RX Complete bit masks the RX Early bit. Whenever RX Complete is set, RX Early will be clear. Also, when the Ethernet controller signals the end of a receive packet, thereby clearing the RX Incomplete bit on the bottom of the RX Status stack, RX Early will be automatically acknowledged or cleared.

It is possible to reprogram the RX Early Threshold value in the middle of receiving a packet and still have the interrupt go off when appropriate. This feature may be used by the software to try to take an interrupt just before its computed packet length in order to overlap some of the packet reception with the expected interrupt latency. To do this, and to generate another interrupt, the RX Early threshold must be reprogrammed *after* the RX Early interrupt has been acknowledged.

Normal collisions may be received as bad packets by the host if the RX Early threshold is set to less than 60 bytes (one slot time). The driver must be prepared to increase this value if too many bad packets result when the setting is less than 60 bytes.

PCMCIA (3C589)

These PC Cards hide 16 bytes of an incomplete packet from the host. Therefore, if RX Early is set to 24, 40 bytes must be received before the packet becomes visible to the host. At that point RX Status shows 24 bytes received. When the packet reception is complete, the 16 bytes will be added to the byte count in the RX Status register all at once. Therefore, the software does not need to be aware of this process.

The power-on default for this threshold is 2,032 bytes.

PCMCIA (3C589B)

The power-on default for this threshold is 2,044 bytes.

The minimum threshold for RX Early is 8. Setting RX Early to less than 8 makes it operate the same as if it were set to 8.

Set TX Available Threshold

1001 0	XXX XXXX XXZZ

This command sets the TX Available threshold, which specifies the number of free bytes required by the host. A TX Available interrupt will be generated when the number of free bytes in the TX FIFO exceeds this threshold. This allows the host to return and perform other processing until there is sufficient free space to continue to copy data to the TX FIFO.

A multiple of four with a range of 0 to 2,044 bytes is used. Setting this threshold to any value greater than 1,792 disables it regardless of the TX FIFO size.

Once the TX Available bit is acknowledged, the threshold returns to default/disabled, and it will not go off again. Therefore, the Set TX Available threshold command must be reissued each time the driver decides to return control until sufficient space is available to continue.

Set TX Start Threshold

1001 1	XXX XXXX XXZZ

This command specifies the number of bytes required in the TX FIFO before the PC Card will start transmitting the packet. The packet will start transmitting either when the number of bytes in the TX FIFO exceeds the argument or when the entire packet has been copied to the TX FIFO. A multiple of four with a range of 0 to 1,792 or above bytes is used. The PC Card truncates the argument to a dword multiple. Setting this argument to any value greater than 1,792 (the power-on default) disables it, so that packet transmission starts only when the entire packet has been moved to the TX FIFO.

Statistics Enable

1010 1	000 0000 0000

This command enables the collection of statistics by the PC Card. These statistics are maintained in Window 6 registers. At power-up, statistics collection is disabled and must be enabled with this command. Before any of the statistics in Window 6 are read, statistics collection must be temporarily disabled with the Statistics Disable command. Once the statistics have been read, statistics can be reenabled with this command.

Statistics Disable

1011 0	000 0000 0000

This command disables the collection of statistics by the PC Card. At power-up, statistics collection is disabled. The PC Card latches statistics update requests while the statistics are disabled. As long as statistics are only kept disabled long enough to read in the statistics, no statistics will be lost in the process. Refer to the "Statistics Registers" section later in this chapter for more information.

Stop Coaxial Transceiver

1011 1	000 0000 0000

This command shuts off the DC-DC converter that drives the on-board coaxial transceiver. This command might be used for diagnostic or power-management reasons. The host must delay at least 800 µs after issuing this command before using the AUI interface. This can be accomplished by starting the timer and waiting until it pegs at its maximum value of FFh.

Power-Management Commands (3C589B)

The following three power-management commands are new with 3C589B PC Cards. They may also be ignored by other future adapters. The capabilities bits can be examined to determine whether power-management capabilities are present in the PC Card.

Power Up*

1101 1	000 0000 0000

This command puts the chip in power-up state, which is the default state at power-on/reset. In power-up state the chip is always fully powered up. This command is typically used after a Power Down Full or Power Auto command. Power Up does nothing if the PC Card is already powered-up. If used after a Power Down Full, the TX Reset and RX Reset commands should be issued before proceeding with operation. See also Power Auto and Power Down Full.

Power Down Full

1110 0	000 0000 0000

This command puts the chip in full power-down state. In this state the only legal access to the chip is the Power Up command. The full power-down state will use the least amount of power, but will not power up automatically. If the DC-DC converter is being used, issuing this command will stop it automatically (the equivalent of Stop Coaxial Transceiver). However, the DC-DC converter must be restarted manually after power-up (with Enable DC Converter, requiring an 800 µs delay). Otherwise, the state of the chip and the contents of its various registers are preserved.

Power Auto

1110 1	000 0000 0000

This command puts the chip in auto-power state. In this state the chip will automatically reduce power when idle (though it will use more power than in the Power Down Full state), and it will return to power-up state automatically when this changes (that is, when an incoming packet starts to be received). After issuing a Power Auto command, the host software should not rely on whether the PC Card is currently powered-up or powered-down. The next access to the PC Card should be a Power Up command. Normally the host software would issue a Power Up command at the entry to any thread dealing with the PC Card (for example, the start of the ISR, or the transmit routine), and issue a Power Auto command before returning. Since the DC-DC converter will not be powered-off with this command, the savings when 10BASE2 is used will be limited, though the command can still be used. The most significant savings will be realized with the 10BASE-T connector, where only a limited amount of logic will remain powered-up (enough to handle the link beat pulse trains, for example).

Status Register

Function: Reports the PC Card state, including window number and the reasons for the interrupt.

Location: All windows/Port 0E

Type: Read only

Size: 16 bits (8-bit reads also allowed to either byte)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Window #			IP	0	0	0	0	US	IR	RE	RC	TA	TC	AF	IL

Figure 6-2 Status Register

Bit Description:		
	Bits 15–13	Window Number (0–7)
	Bit 12	Command-in-Progress
	Bit 11	Reserved
	Bit 10	Reserved
	Bit 9	Reserved
	Bit 8	Reserved
	Bit 7	Update Statistics†
	Bit 6	Interrupt Requested†
	Bit 5	RX Early†
	Bit 4	RX Complete†
	Bit 3	TX Available†
	Bit 2	TX Complete†
	Bit 1	Adapter Failure†
	Bit 0	Interrupt Latch

Those bits marked with a dagger (†) cause an interrupt when set, assuming they are not masked off in either the Interrupt mask or the Read Zero mask. These bits can be forced to zero with the Read Zero mask, disabled as a source of interrupts with the Interrupt mask, or acknowledged with the Acknowledge Interrupt command. All of these have identical layouts for these bits.

The low byte contains all of these interrupt causes and can be used directly as an index into a dispatch table if desired.

PCMCIA (3C589B)

A Host Reset (Global Reset with Host Reset enabled) clears the interrupt bits but does *not* clear the interrupt source. Therefore, if the Read Zero mask and the Interrupt mask are set, the interrupt may reappear. The driver must reset the interrupt in order to clear it.

Bit 0 Interrupt Latch This bit is latched when the PC Card raises an interrupt to the host. The bit is cleared when it is acknowledged. In a shared interrupt environment (not ISA), this bit can be used to determine the source of the interrupt.

*This bit does not cause an interrupt, it **is** the interrupt.*

Bit 1 Adapter Failure† An error occurred that the PC Card was unable to recover from. Possible causes are:

- Transmit overrun (host writes more data than there is room for)
- Receive underrun (host reads data that is not yet available or attempts to read the RX FIFO beyond the pad bytes)
- Internally detected hardware errors as yet undefined

Possible causes are described in the FIFO Diagnostic Port register. The host must issue the appropriate Reset command to clear this condition and recover.

Bit 2 TX Complete† The PC Card has finished transmitting a packet and has updated the TX Status register with its transmit status (with the TX Complete bit set). To clear this bit, the host writes the TX Status register to pop the transmit status off the TX Status stack.

A TX Complete interrupt is signaled only for packets that failed to transmit successfully or packets with a transmit preamble bit set specifically for an interrupt on successful transmission.

If TX Status indicates a transmit underrun or jabber error, then a TX Reset command will be necessary. In any event, a TX Enable command will be required to restart the transmitter after any error. Refer to "TX Status" later in this chapter for more information.

Bit 3 TX Available† The number of free bytes in the TX FIFO now exceeds the TX Available threshold. Clear this bit using the Acknowledge Interrupt command.

Bit 4 RX Complete† A complete packet is available in the RX FIFO. This bit is set if the Incomplete bit in the RX Status register is zero. The RX Status register can now be examined for possible errors and packet length.

To clear this bit, the host must read the packet out of the RX FIFO. A receive overrun requires no special action on the part of the host, other than discarding the packet.

Bit 5 RX Early† Sufficient bytes of the current packet have been received to exceed the RX Early threshold, although the packet is not yet complete. See the Set RX Early Threshold command for more details. To clear this bit, use the Acknowledge Interrupt command, or wait for RX Complete.

Bit 6 Interrupt Requested† This bit is set by the Request Interrupt command. To clear this bit, simply acknowledge it. It provides a way for the driver to request an interrupt for its own purposes.

Bit 7 Update Statistics† This bit indicates that one or more of the statistics counters are nearing an overrun condition (typically half a counter's maximum value). The host must read out all of the statistics and update its local counters from them (zeroing the counters on the PC Card in the process and thereby clearing this bit).

Bit 12 Command-in-Progress

This bit is set to indicate that the last command issued is still being processed by the PC Card. It need be checked only after one of the commands has been issued that may require more than a single I/O cycle for completion (these commands are marked with an asterisk in the Command Register explanation earlier in this chapter). No other commands may be issued until this bit has been reset. This check must be done with interrupts disabled.

Rereading the EEPROM takes about 250 microseconds. This occurs after a power-up reset, a CC Reset, or an AISM Reset.

Bits 15–13 Window Number

These bits reflect the current window set and must be visible in every window. The window number is reset to zero at power-up or after a Host Reset.

FIFO Registers

RX Status

Function: Contains the status and number of bytes for the receive packet at the top of the RX FIFO.

Location: Window 1/Port 08

Type: Read only

Size: 16 bits

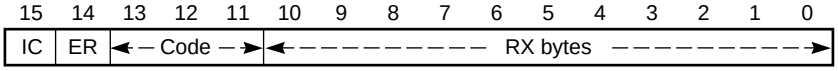


Figure 6-3 FIFO Register

Bit Description:

Bit 15	Incomplete (1 = RX packet is incomplete or RX FIFO empty)
Bit 14	Error (1 = error in RX packet, 0 if incomplete or no error)
Bits 13–11	**Type of error (undefined if no error, highest priority first)** **000 = Overrun** **011 = Runt Packet Error** **100 = Alignment (Framing) Error** **101 = CRC Error 001 = Oversize Packet Error (>1,514 bytes or >1,518 bytes if Disable CRC Generation set).** **If bit 14 = 0, then 010 = Dribble Bit(s) information only or all other codes and 0 = no errors**
Bits 10–0	RX Bytes (0–1,514)

This register is a ripple-through FIFO that advances one position (popping the stack) after issuing an RX Discard command. As a packet is received off the wire, its corresponding RX Status entry (visible if this is the only packet in the RX FIFO) has its RX Bytes field incremented. Until the end of the packet is received and placed in the RX FIFO, the Incomplete bit will remain set and the Error bit will be clear. Once the packet has been completely moved into the RX FIFO, the Incomplete bit will be cleared and the Error bit and the error type will be set appropriately.

The error bits encode the packet status. If multiple errors have occurred, the highest-priority error will be shown. For example, a runt with a CRC error will be flagged as a runt. If a packet is flagged as overrun, the host must not rely on the contents of the packet (it may have holes in it where the data was lost).

A packet with only a Dribble Bit error is a valid packet and must be read by the driver software. The Dribble Bit indication is for informational purposes only.

An oversized packet (longer than 1,514 bytes, or 1,518 if Disable CRC Generation is set in the Media Type and Status register) will continue to be received correctly until it reaches 1,792 bytes, when it will be cut off and the remainder discarded.

If the packet is not read from the RX FIFO until the Incomplete bit is cleared, then the RX Bytes field will show the packet length, assuming there were no errors.

As the packet is read from the RX FIFO, RX Bytes will be decremented. This can be done before the packet is completely received; in this case, RX Bytes will not show the actual packet size.

RX Bytes reflects the number of bytes of packet data that were received, and does not include any padding to a dword multiple.

A packet becomes visible to the host through RX Status once the number of bytes received exceeds the minimum of 60 and the RX Early threshold. At that point, the number of bytes received becomes visible in RX Bytes and will continue to increment as more bytes are received. If an error in the packet is signaled before the packet becomes visible to the host, the packet is discarded. Otherwise, the packet will show up in the RX Status stack flagged with an error.

While the last byte of the actual packet data is being read, RX Bytes will show 1. If the packet was not a multiple of four in length, there may be more bytes to read because of padding. After the last byte has been read, RX Bytes will change to 0. After one more byte is read, RX Bytes will change to –1 (11111111111), then –2 (11111111110) after one more byte, then –3. Software must not attempt to read beyond the pad bytes (an RX Discard command must be issued to access the next packet, if any).

PCMCIA (3C589)

These PC Cards hide 16 received bytes from the host, so that the RX Bytes count will always be 16 less than the number of bytes received off the wire until the packet is completed. At that point, RX Bytes will be incremented by 16 bytes.

TX Status

Function: Reports the transmit status of a completed transmission. Writing this register pops the transmit completion stack.

Location: Window 1/Port 0B

Type: Read only

Size: 8 bits

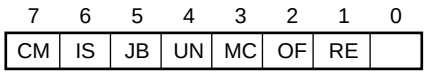


Figure 6-4 TX Status Register

Bit Description:

Bit 7	Complete (1 = TX is complete)
Bit 6	Interrupt on Successful Transmission Completion Requested
Bit 5	Jabber Error (TP only: TX Reset required)
Bit 4	**Underrun (PCMCIA [3C589]: TX Reset required)** **(PCMCIA [3C589B]: Network TX Reset and FIFO TX Reset required)**
Bit 3	Maximum Collisions
Bit 2	TX Status Overflow
Bit 1	Undefined (PCMCIA [3C589, 3C589B])
Bit 0	Undefined

The hardware uses the TX Status register to stack information about transmit completions that must be signaled to the driver. Whenever a transmit is completed that must be signaled to the host (either it failed, or the preamble specified an interrupt on successful transmission), the PC Card pushes the status onto the TX Status stack. When the host fields the TX Complete interrupt, it can read TX Status to determine the transmit status. The TX Complete bit will not be set if the stack is currently empty (nothing to pop). Whenever the driver writes the TX Status register and the TX Complete bit is set, this pops the stack, and the next transmit complete status can be read (if any). Popping everything off the TX Status stack turns off the TX Complete interrupt in the Status register. Do not write to the TX Status register unless you have read a value with the TX Complete bit set. To do so may clear a yet-to-be seen transmit status.

If an error is indicated, then the transmitter has been disabled and must be reenabled with the TX Enable command. If the error was a maximum collisions error, then nothing more is required. If the error was a jabber or an underrun, however, then a TX Reset command is required before the TX Enable can be issued. To recover from an underrun, the host should issue the TX Reset (Non-Immediate) command to guarantee that the packet goes out with a bad CRC.

When the completion of a packet is signaled to the host, the packet has been discarded from the TX FIFO. If it is to be retransmitted, it must again be copied to the TX FIFO. If the error occurred while the packet was still being copied to the PC Card, the host can continue to copy the packet to the PC Card, since the transmitter is disabled.

Free Transmit Bytes may take an additional I/O cycle to update after the packet transmission has been completed.

The Interrupt on Successful Transmission Requested bit reflects the same bit in the TX preamble for this packet. The protocol can use this bit to determine whether this is the packet on the head of some "to be completed" queue, or simply a packet that it has forgotten. In either case the host can use this opportunity to update any statistics counters it may have.

The TX Status Overflow bit, if set, indicates that the TX Status stack is full, and as a result the transmitter has been disabled. Writing the TX Status register clears this condition; no other action is required. The TX Status stack can hold exactly 31 entries, so this condition is unlikely in normal operation. No packets are dropped or confirmations lost when this condition is entered.

RX PIO Data Read

Function: Used to read data from the RX FIFO.

Location: Window 1/Port 00 and Port 02

Type: Read only

Size: 32 bits/16 bits/8 bits allowed from lower byte.

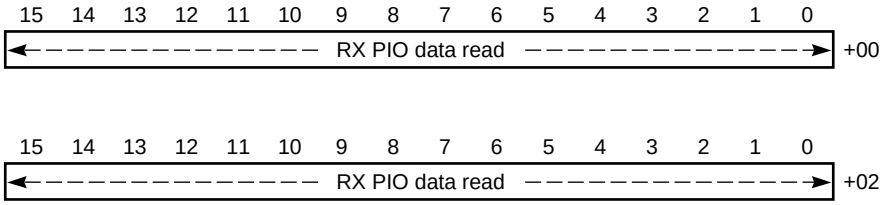


Figure 6-5 RX PIO Data Read Register

Bit Description: Bits 15–0 RX PIO Data Read

PCMCIA (3C589, 3C589B)

This register is used to read data from the RX FIFO. A word read pops a word off the head of the RX FIFO. A byte read to the low-order byte of the register pops a byte off the RX FIFO. Byte reads to the high-order byte are not allowed. Double word reads are also possible when this register is used in combination with the one after it (which is treated identically by the hardware). Such reads pop two successive words off the RX FIFO, returning the first in the low-order word, and the second in the high-order word.

Although byte and word reads are allowed, the packet data is always padded to a dword boundary. The driver must not attempt to read the FIFO beyond the pad bytes. The RX Discard command can be used to skip this padding.

TX PIO Data Write

Function: Used to write data to the TX FIFO.

Location: Window 1/Port 00 and Port 02

Type: Write only

Size: 32 bits/16 bits/8 bits allowed to lower bytes

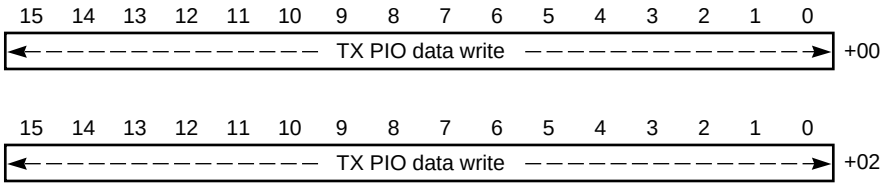


Figure 6-6 TX PIO Data Write Register

Bit Description: Bits 15–0 TX PIO Data Write

PCMCIA (3C589, 3C589B)

This register is used to write data to the TX FIFO. A word write pushes a word onto the tail of the TX FIFO. A byte write to the low-order byte of the register pushes a byte onto the TX FIFO. Byte writes to the high-order byte are not allowed. Double word writes are also possible when this register is used in combination with the one after it (which is treated identically by the hardware). Such writes push two successive words on the TX FIFO; the low-order word first, then the high-order word.

Although byte and word writes are allowed, the packet data must always be padded to a dword boundary.

Free Receive Bytes

Function: Returns the number of bytes available in the RX FIFO.

Location: Window 3/Port 0A

Type: Read only

Size: 16 bits

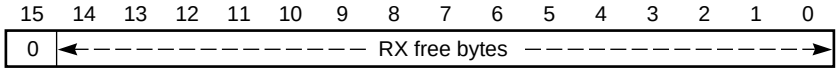


Figure 6-7 Free Receive Bytes Register

Bit Description: Bits 14–0 Free Receive Bytes (0–32,764)

The actual range depends on the PC Card and configuration of the RX FIFO. To determine the actual RX FIFO size, read this register subsequent to RX Reset.

Free Receive Bytes equal to 0 imply that the RX FIFO is full.

Free Transmit Bytes

Function: Returns the number of bytes available in the TX FIFO.
Location: Window 1/Port 0C and Window 3/Port 0C
Type: Read only
Size: 16 bits

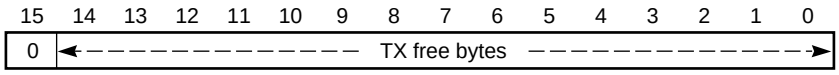


Figure 6-8 Free Transmit Bytes Register

Bit Description: Bits 14–0 Free Transmit Bytes (0–32,764)

The actual range depends on the PC Card and configuration of the TX FIFO. To determine the actual TX FIFO size, read this register subsequent to TX Reset.

The value in Window 1 is truncated to a dword multiple to assist dword-aligned data transfers. The value in Window 3 is the actual number of free bytes.

Free Transmit Bytes equal to 0 imply that the TX FIFO is full. When Free Transmit Bytes are not zero, that number of bytes can be written to the TX FIFO without an overrun.

A small number of bytes per packet are required for hardware overhead. A driver should not depend on the exact number of bytes shown in TX/RX Free to compute the packet size.

Timer Register

Function: Interrupt latency measurement.
Location: Window 1/Port 0A
Type: Read only
Size: 8 bits

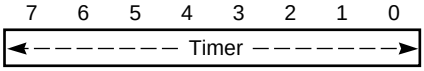


Figure 6-9 Timer Register

Bit Description: Bits 7–0 Timer

The Timer register is a free-running 8-bit counter that is running off a 10 MHz/32 clock period (period = 3.2 µs, the quadbyte rate). The count is reset to zero whenever the interrupt output transitions from inactive to active. This allows the driver to make interrupt latency measurements and evaluate other system-dependent parameters. When the Timer bit reaches 255, it stops incrementing.

To use this counter for more general measurements at driver initialization time, clear the Interrupt Latch, disable interrupts, and issue a Request Interrupt command to start the counter.

Statistics Registers

The Statistics registers in Window 6 make available various counters maintained
by the PC Card. Before reading any of these registers, the host must disable statistics collection by issuing the Statistics Disable command. Reading any statistic resets
it to zero. Counters that are 16 bits in the interface must be read as words. Counters that are eight bits must be read as bytes. After the counters have been read, the Statistics Enable command must be used to reenable statistics collection.

The size of the internal counters on the PC Card varies. Several counters are six bits, and several are only four bits. Counters less than eight bits long get zero fill to eight bits across the interface. An Update Statistics interrupt is generated when any statistic reaches half its maximum value (that is, its upper bit gets set). The exceptions are the two 16-bit counters that generate an Update Statistics interrupt only when the upper three bits are set (so they can use more of their range before they require reading).

The proper method for servicing this interrupt is to read all statistics and to accumulate proper 32-bit values in host memory. This will reset each statistic in turn and update all statistics in a relatively coherent manner. There is no bit available to indicate which counter reached half its value. Once all of the counters have been read, the Update Statistics bit in the Status register will be cleared. The PC Card latches statistics update requests while the statistics are disabled. As long as statistics are kept disabled only long enough to read the statistics in, no statistics will be lost in the process.

Writing to the Statistics registers is supported to allow for simple debugging. When a value is written to the Statistics registers, it is added to the current value for that statistic. This is done using much of the normal Statistics Collection state machine. Statistics collection must be disabled for the addition to take place.

The following network statistics are defined primarily according to the NDIS specification, version 2.01.

Statistics

Function:	Read and zero internal counters.
Location:	Window 6/Ports 00 through 0C
Type:	Read/write (reading resets counter)
Size:	8 bits/16 bits (16 bits as words, all else as bytes)

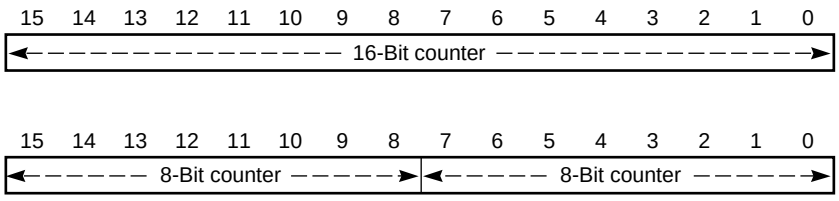


Figure 6-10 Statistics Register

These counters collect statistics of transmit or receive events as long as statistics are currently enabled. Statistics must be disabled before they can be read. Reading a statistic also zeroes it. To check proper function of the statistics logic, values can also be written to the statistics.

Writing a value adds that value to the current statistic as long as statistics are currently disabled. *The 16-bit statistics must be read/written as a word; all others must be read/written as bytes. Unpredictable results occur if these rules are not followed.*

FIFO loopback can cause the various Statistics registers to return unreliable values. After leaving FIFO loopback, enable statistics; then disable and read all the statistics to clear them.

Bit Description: Bits 15–0 or 7–0 Counter value

Table 6-1 Statistics in Statistics Register

Port Offset	Size (bits)	Description of Statistic
0C	16	Total bytes transmitted successfully with no errors noted.
0A	16	Total bytes received successfully. This number excludes runts, overruns, and frames discarded before completion.
08	8	Total transmit deferrals.
07	8	Total frames received successfully. This number excludes runts, overruns, and frames discarded before completion.
06	8	Total frames transmitted successfully with no errors noted.
05	8	Total receive frames discarded because of RX FIFO overrun. This includes only those packets seen by the host as RX overruns. It does not include those discarded without a trace because the RX FIFO was completely full.
04	8	Total late collisions on transmit.
03	6	Total frames transmitted after one collision.
02	6	Total frames transmitted after multiple collisions.
01	4	Total frames transmitted with no CD heartbeat (SQE). This statistic is only collected if the SQE Statistic Enable bit (bit 3 of the Media Type and Status register) is set. Since certain external transceivers do not support SQE, this statistic can be disabled to avoid excessive update statistics.
00	4	Total carrier sense lost during transmission.

With 3C589 PC Cards, if you get an Update Statistics interrupt when the transmitter is in error (TX Status has TX Complete set), then do not read the late collisions, carrier sense lost, or deferrals statistics. These statistics may be incrementing at a high rate. Instead, disable the Update Statistics interrupt with the Read Zero mask. Once the error has been handled or a TX Reset issued, read these three statistics to zero and reenable. 3C589B PC Cards do not require the Read Zero mask for the interrupt. All EtherLink III adapters require the Statistics command to be issued before the statistics are read.

Diagnostic Registers

Media Type and Status

Function: Reports media type/configuration and status.
Location: Window 4/Port 0A
Type: Read/write (only certain bits are writable)
Size: 16 bits

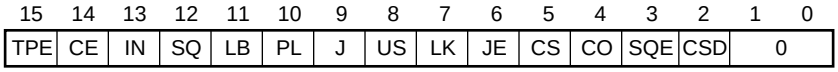


Figure 6-11 Media Type and Status Register

Bit Description:

Bit 15	1 = TP (10BASE-T) enable (read only).
Bit 14	1 = Coax (10BASE2) transceiver enabled (read only).
Bit 13	1 = Reserved, always 1 (read only).
Bit 12	1 = SQE present (read only).
Bit 11	1 = Valid link beat detected (TP) (read only).
Bit 10	1 = Polarity reversal detected (TP) (read only).
Bit 9	1 = Jabber detected (TP) (read only) (see TX Status).
Bit 8	1 = Unsquelch (TP) (read only).
Bit 7	1 = Link beat enabled (writable). Defaults to 0 (disabled). Must be set by software only if the internal TP transceiver is in use. Link beat must be disabled in ENDEC loopback mode.
Bit 6	1 = Jabber enabled (writable). Defaults to 0 (disabled). Must be set by software only if the internal TP transceiver is in use. This also enables the Polarity Reversal state machine.
Bit 5	1 = Carrier sense (CRS) (read only).
Bit 4	1 = Collision (read only).
Bit 3	1 = SQE Statistics Enable. Defaults to 0 (disabled). Must be enabled by software at startup if AUI is selected, and disabled only if the number of SQE errors becomes excessive in AUI mode, which probably indicates that the external transceiver does not support SQE.

Bit 2 — 1 = CRC Strip Disable (3C509B and 3C589B). Bridging software can determine whether this function is supported by attempting to set the bit and checking to see if the bit changes to a one. If this bit does not change to a one, do not assume that this function or the pass-through CRC will be supported. Defaults to 0 (CRC stripping enabled, no pass-through) at power-up/ reset. Setting this bit allows the host software to receive the receive frame's 4-byte CRC as part of the packet data (and have it be counted in the RX Bytes count and the Bytes Received OK statistic). It will indirectly change the definition of an oversized receive frame from one greater than 1,514 bytes, to one greater than 1,518 bytes. The PC Card will continue to check the CRC being received and flag those in error through RX Status. This bit may be set by bridging software looking to pass through the CRC in order to guarantee that any local memory errors will result in a bad packet. A packet can be transmitted with such a pass-through CRC by setting the Disable CRC Generation bit in the transmit preamble when transmitting it.

Bits 1–0 — Unassigned, read as zero.

Net Diagnostic Port

Function: Supports the network diagnostic.

Location: Window 4/Port 06

Type: Read/write (only certain bits are writable)

Size: 16 bits

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EL	ENL	ECL	FL	TXE	RXE	TXT	TXR	SE	0	ASIC					TLD

Figure 6-12 Net Diagnostic Port Register

Bit Description:

Bit 15 — 1 = External loopback (read/write). Defaults to zero (disabled). Setting this bit enables an external loopback mode allowing simultaneous transmit and receive (TP, BNC, or AUI external loopback mode).

Bit 14 — 1 = ENDEC loopback (read/write). Defaults to zero (disabled). Setting this bit enables a loopback mode at the output of the encoder/decoder. You must disable the link beat through the Media Type and Status register before enabling ENDEC loopback.

Bit 13	1 = Ethernet controller loopback (read/write). Defaults to zero (disabled). Setting this bit enables loopback at the output of the Ethernet controller.
Bit 12	1 = FIFO loopback (read/write). Defaults to zero (disabled). This loopback returns data through the FIFO at the interface between the Ethernet controller transmitter and the FIFO. In FIFO loopback mode, overruns and underruns are not possible—the data is simply moved between the FIFOs as it is available.
Bit 11	1 = TX enabled (read only). Can be cleared by TX Reset, TX Disable, or as a result of a transmit error.
Bit 10	1 = RX enabled (read only).
Bit 9	1 = TX transmitting (read only). Set if the transmitter is transmitting or deferring before transmitting.
Bit 8	1 = TX Reset required (read only). Set if a jabber or underrun error occurs, both of which require a TX Reset for recovery.
Bit 7	1 = Statistics enabled (read only).
Bit 6	Unassigned, read as zero.
Bits 5–1	ASIC revision level (read only) 3C589 =1. 3C589B= 2. Possible future revisions of the ASIC that change the functionality in any significant way will modify this value.
Bit 0	1 = Test low-voltage detector (write only). Setting this bit to one will reset the ASIC if the low-voltage detector is functional. This bit defaults to zero at power-up/reset. This bit must remain zero except for ASIC functional testing.

FIFO loopback can cause the various Statistics registers to return unreliable values. After leaving FIFO loopback, enable statistics; then disable and read all the statistics to clear them.

FIFO Diagnostic Port

Function:	Supports the FIFO diagnostics.
Location:	Window 4/Port 04
Type:	Read/write (only certain bits are writable)
Size:	16 bits

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
RXR	RSD	RXU	RSO	RXO	TXO	0	0	RSD							

Figure 6-13 FIFO Diagnostic Port Register

Bit Description:	Bit 15	1 = RX receiving (read only). Set when a packet is being received into the RX FIFO.
	Bit 14	Reserved.
	Bit 13	1 = RX Underrun (read only). Generates Adapter Failure interrupt. Requires RX Reset or Global Reset command to recover. An RX Underrun is generated only when you read past the end of a packet—reading past what has been received so far will give bad data.
	Bit 12	1 = RX Status Overrun (read only). Set when there are already eight packets in the RX FIFO. While this bit is set, no additional packets are received. Requires no action on the part of the host. The condition is cleared once a packet has been read out of the RX FIFO. For PCMCIA (3C589B), this bit is reserved and read as zero.
	Bit 11	1 = RX Overrun (read only). Set when the RX FIFO is full (there may not be an overrun packet yet). While this bit is set, no additional packets will be received (some additional bytes can still be pending between the wire and the RX FIFO). Requires no action on the part of the host. The condition is cleared once a few bytes have been read out of the RX FIFO.
	Bit 10	1 = TX Overrun (read only). Generates Adapter Failure interrupt. Requires the TX Reset or Global Reset command to recover. Disables transmitter.
	Bits 9–8	Unassigned, read as zero.
	Bits 7–0	For 3C589 these bits are used to execute the Built-in Self-test (BIST) circuitry for both the RX and TX FIFOs. These bits will give 100% fault coverage for stuck-at faults, transition faults, coupling faults, and addressing (decoder) faults. They are intended primarily for testing the ASICs, but can also be included in a diagnostic self-test. The two tests, RX and TX, are run independently. To perform either test, first reset, then set the appropriate BIST bit. Loop until the BC is set (this takes approximately 500 ms). If BF is set, the test failed. Otherwise check for BF stuck at 0. To do this, set BFC, then reset it. BF must now be set. If not, BF is stuck. If everything passes, then the RAM is fully functional. All read-only bits default to reset (0) at power-up. For 3C589B, these bits are reserved, undefined.

Bit 7 RX BIST (write only). Enables the BIST embedded in the RX FIFO RAM.

Bit 6 RX BFC (write only). Unconditionally sets RX BF (used to check stuck-at faults on RX BF). This check should be done after the successful completion of a BIST to ensure proper operation of the RX BF.

Bit 5 RX BF (read only). Indicates the BIST has failed. It is also set by RX BFC. RX BF is only valid if RX BFC or RX BC is set.

Bit 4 RX BC (read only). Indicates the BIST is complete.

Bit 3 TX BIST (write only). Enables the BIST embedded in the TX FIFO RAM.

Bit 2 TX BFC (write only). Unconditionally sets TX BF (used to check stuck-at faults on the TX BF). This check should be done after the successful completion of a BIST to ensure proper operation of TX BF.

Bit 1 TX BF (read only). Indicates the BIST has failed. It also can be set by TX BFC. TX BF is only valid if TX BC or TX BFC is set.

Bit 0 TX BC (read only). Indicates the BIST is complete.

Ethernet Controller Status

Function: Provides access to Ethernet controller status.

Location: Window 4/Port 08

Type: Read/write (only certain bits are writable)

Size: 16 bits

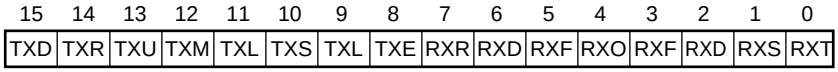


Figure 6-14 Ethernet Controller Status Register

The writable bits all default to zero at power-up/reset. They must remain zero except when writing test vectors.

Bit Description:

Bit 15	Ethernet controller TX DONE (read only).
Bit 14	Ethernet controller TX RETRY (read only).
Bit 13	Ethernet controller TX UNDERRUN (read only).
Bit 12	Ethernet controller TX MAX COLL (read only).
Bit 11	Ethernet controller TX LATE COLL (read only).
Bit 10	Ethernet controller TX SQE ERR (read only).
Bit 9	Ethernet controller TX LCAR (read only).
Bit 8	Ethernet controller TX END SLT TIME (read only).
Bit 7	Ethernet controller RX REJECT OUT (read only).
Bit 6	Ethernet controller RX DONE (read only).
Bit 5	Ethernet controller RX FRAME ERR (read only).
Bit 4	Ethernet controller RX OVERRUN (read only).
Bit 3	Ethernet controller RX FCS ERR (read only).
Bit 2	Ethernet controller RX DRIBBLE (read only).
Bit 1	Ethernet controller RX SHORT (read only).
Bit 0	Ethernet controller RX TESTEN (read/write)

7 PC CARD CONFIGURATION AND ENABLE

The 3C589 PC Cards support a variety of configuration options (such as IO Base Address, ROM Size, and Interrupt Level) that are controlled by a set of configuration registers.

The PC Cards support the PCMCIA activation mechanism

Once activated, the PC Card can be deactivated only by a Global Reset command.

Automatic Configuration at Power-on Reset

The driver software is responsible for reading the station address out of the EEPROM and writing it into the appropriate registers in Window 2.

For the 3C589 PC Card, after reset (power-on reset, the Global Reset command, or soft reset with the PCMCIA Configuration Control register bit 2), auto configuration is not run. The driver software must first enable the PC Card's I/O interface and then write the appropriate data to the following registers: Address Configuration, Resource Configuration, and, optionally, Product ID.

PCMCIA Activation Mechanism (3C589 PC Card)

The 3C589 hardware supports the PCMCIA-specified architecture, allowing PCMCIA Card Services basic management of PC Card resources. The PC Card provides standardized data structures and registers such as the Card Information Structure, the Configuration Option register, and Card Configuration and Status register. Drivers access this data structure and registers through Card Services.

After a power-on or a soft reset, the PCMCIA PC Card appears to the host as a memory-only device, and Window 0 is selected as the working register set. Driver software should use PCMCIA Card Services to find the PC Card, allocate resources such as the I/O base and interrupt request level, and enable the PC Card's I/O interface. The driver should save the I/O base and IRQ values for subsequent PC Card operation. See the PCMCIA 2.xx specification for initialization details and the programming interface to Card Services.

Once the PC Card's I/O interface is enabled, the PC Card's Address Configuration register and Resource Configuration registers must be programmed. In particular, the ROM SIZE, ROM BASE, and I/O BASE fields of the Address Configuration register must each be set to zero and the IRQ field of the Resource Configuration register must be set to 3. (The actual I/O base and IRQ used during driver operation are those obtained from PCMCIA Card Services, above.)

Refer to the sections "PCMCIA-Specific Data Structures" later in this chapter for more information.

Configuration Overview (3C589)

The PC Card is a controller that interfaces the PCMCIA sockets with a system. For example, the socket's single IRQ line is mapped to one of the 15 IRQs supported by an ISA system. The PCMCIA PC Card uses the dedicated resources of the slot where it is installed. Therefore, the configuration register bits that normally control how the PC Card assigns resources no longer apply and are in general ignored. The resources of a PCMCIA PC Card are assigned ISA resources by a programming mechanism. This is done directly via hardware; alternatively, it can be done via software using either a point enabler or Socket Services. How this is done is beyond the scope of this document.

For I/O accesses, the PCMCIA PC Card decodes the bottom four bits of the address, treating the upper bits as "don't cares." Under PCMCIA there are two types of memory: attribute and common memory. The CIS storage and some PCMCIA-specific configuration registers are in attribute memory. The common memory maps the ROM.

As far as the PC Card is concerned, there is no CIS/ROM boundary in attribute memory. CIS storage simply starts at offset 0 in the ROM and ends somewhere in the first 64 K. If attribute memory is being addressed and A[16] is on, then a register is being addressed, and A[1] determines which register it is. Any other memory reference, whether attribute or common, simply addresses the ROM. CIS storage will be located at the start of the ROM, with any other data starting at the next 4 K boundary after CIS.

PCMCIA Configuration Option Register

The Configuration Option register is used to configure the card and to issue a soft reset to the card. The register is a read/write register that contains three fields, as shown in Figure 7-1 and Figure 7-2.

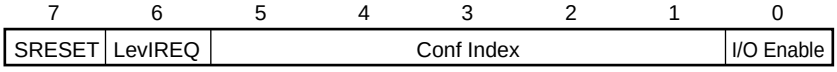


Figure 7-1 Configuration Option Register (3C589)

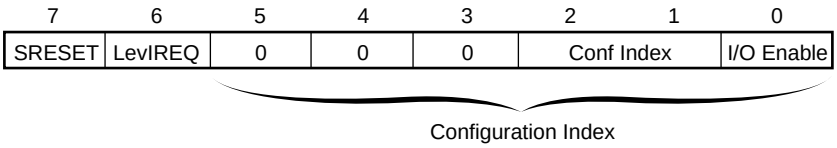


Figure 7-2 Configuration Option Register (3C589B)

SRESET — Soft RESET Card. Setting this bit to one places the card in the reset state. This is equivalent to assertion of the +RESET signal except that this bit is not cleared. Returning this bit to zero leaves the card in the same unconfigured reset state as following power-up or hardware reset. This bit is set to zero by power-up and hardware reset. This bit should not affect the modem attribute registers, clock, or other bits that relate to the modem.

LevIREQ — Level Mode interrupts are selected when this bit is one. Pulse Mode interrupts are selected when the bit is zero.

Reserved — Reserved bits must be zero.

Conf Index — Configuration Index. Used to indicate which configuration of a set has been chosen for the card.

3C589 Bits 1–5 Conf Index
3C589B Bits 1–2 Conf Index
3C589B Bits 3–5 Always zero

I/O Enable — When set to zero, the card's I/O is disabled; it will not respond to any I/O cycles and will use the Memory-Only interface. Setting this bit to one enables the I/O interface.

Changes for 3C589D

SRESET — When this bit is set, all LAN functions are reset as if the card was given a hardware reset. This bit should not affect the modem attribute registers, clock, or other bits that relate to the modem.

LevIREQ — If LAN has interrupts enabled on IREQ line(see bit 2 below) and this bit is set, the IREQ line will have level mode interrupts enabled.

Reserved — Reserved bits must be zero.

Conf Index — Bit 5-4: Always zero

Bit 3: Status Change Mode:

When set, the LAN interrupt is passed out (in level mode only)on the STSCHG line. Modem STSCHG events will be blocked in order to avoid missing an interrupt from the LAN. The INTR bit of the LAN CSR will not reflect the state of the LAN interrupt since any software that relies on this bit will not be aware of the STSCHG interrupt mechanism.

Bit 2: Enable IREQ Routing

If only LAN is enabled (MODEM COR bit 0 is disabled), then LAN interrupts are always on IREQ line. If both functions are enabled, LAN interrupts are only put on the IREQ line is this bit is enabled.

Bit 1: Enable Base/Limit Registers

When set, LAN will respond only to I/O accesses in the 16 byte range starting with the address in LAN IOBASE0 and LAN IOBASE1. If LAN COR bit 0 is enabled, and MODEM COR bit 0 is disabled, and this bit is disabled, the LAN will respond to all I/O accesses, regardless of address. (This is LAN-Only mode).

I/O Enable — When set, LAN is enabled and card is placed in I/O mode. When not set, LAN will ignore all I/O accesses.

PCMCIA Card Configuration and Status Register

The Card Configuration and Status Register contains information about the card's condition. Refer to Figure 7-3 and Figure 7-4.

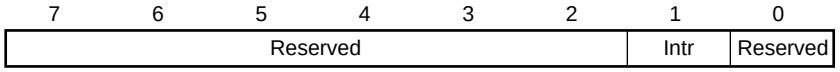


Figure 7-3 Configuration and Status Register (3C589)

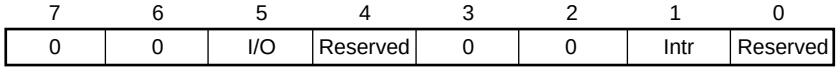


Figure 7-4 Configuration and Status Register (3C589B)

Reserved: Reserved bits must be zero.

Intr: This bit represents the internal state of the interrupt latch. This value is available whether or not interrupts have been configured. This signal remains true until the condition that caused the interrupt has been serviced.

3C589B

Bit 7 Changed. Always zero.

Bit 6 Signal Change. Always zero.

Bit 5 I/O. Set to one when the host can provide only an 8-bit data path. The PC Card then responds as an 8-bit device.

Bit 4 Reserved.

Bit 3 Audio. Always zero.

Bit 2 Power Down. Always zero.

Bit 1 Intr. Same as for PCMCIA 3C589.

Bit 0 Reserved.

Changes for 3C589D

Bit 5 IOIs8: When set, the LAN will not assert the PC_IOIS16 line during I/O cycles. this will force the host machine to perform two 8 bit accesses to read the I/O port.

Bit 2 PowerDown: When set, the LAN portion of the card will enter a low-power state.

Bit 1 Intr: This bit is "1" when the LAN has interrupts enabled on the IREQ line and is currently requesting interrupt servicing.

Bit 0 IntrACK: If all functions that have interrupts enabled on the IREQ line have this CSR bit set, the card is placed in the "IntrACK" mode which requires the host to write a "0" to either Intr bit upon servicing the card interrupt. See MutiFunction Spec for details.

Changes for 3C589D The following are added registers for 3C589D:

LAN IOBASE 0 When the LAN COR bit 1 is set, only I/O accesses in the 16 bytes above and including LAN IOBASE[11:4] will be answered. Otherwise, this register is ignored .

7	6	5	4	3	2	1	0
IOBASE7	IOBASE6	IOBASE5	IOBASE4	0	0	0	0

Figure 7-5 LAN IOBASE 0

LAN IOBASE 1

7	6	5	4	3	2	1	0
0	0	0	0	IOBASE11	IOBASE10	IOBASE9	IOBASE8

Figure 7-6 IOBASE 1

MODEM COR

7	6	5	4	3	2	1	0
SRESET	Level Int	R/W	R/W	R/W	En IREQ	Base/Limit	Enable

Figure 7-7 Modem COR

Bit 7 SRESET: When this bit is set, all Modem functions are reset as if the card was given a hardware reset. This bit should not affect any parts of the device not related to the modem.

Bit 6 Level Interrupt: If Modem has interrupts enabled on IREQ line (see bit 2 below) and this bit is set, the IREQ line will have level mode interrupts enabled.

Bit 5-3 R/W: These bits have no function but should be host read-and-writable.

Bit 2 Enable IREQ Routing: If only Modem is enabled (LAN COR bit 0 is disabled), then Modem interrupts are always on IREQ line. If both functions are enabled, Modem interrupts are only put on the IREQ line is this bit is enabled.

Bits 1 Enable Base/Limit Registers: When set, Modem will respond only to I/O accesses int he 8-32 byte range (depending on I/O limit) starting with the address in MODEM IOBASE0 and MODEM IOBASE1. If MODEM COR bit 0 is enabled, and LAN COR bit 0 is disabled, and this bit is disabled, the MODEM will respond to all I/O accesses, regardless of address. (This is Modem-Only mode).

Bit 0 Enable Function: When set, Modem is enabled and card is placed in I/O mode. When not set, Modem will ignore all I/O accesses.

MODEM CSR

7	6	5	4	3	2	1	0
Changed	SigCh	IOIs8	0	Audio	PowerDown	Intr	IntrAck

Figure 7-8 Modem CSR

Bit 7 Changed: A status change event has occurred.

Bit 6 SigChange: When set, ASIC will assert STSCHG upon a status event.

Bit 5 IOIs8: When set, ASIC will not assert the PC_IOIS16 line during Modem I/O cycles. This will force the host machine to perform two 8 bit accesses to read the I/O port.

Bit 4 Reserved

Bits 3 Audio: When the modem is active, the COLL pin becomes AUDIO (an audio enable bit). This output will reflect the Audio bit in the CSR.

Bit 2 PowerDown: When set, the Modem portion of the card will enter a low-power-state. When the MoWEout bit is clear, the WE PCMCIA strobe becomes the PowerDown output which has programmable polarity.

Bit 1 This bit is "1" when the Modem has interrupts enabled on the IREQ line and is currently requesting interrupt servicing.

Bit 0 IntrACK: If all functions that have interrupts enabled on the IREQ line have this CSR bit set, the card is placed in the "IntrACK" mode which requires the host to write a "0" to either Intr bit upon servicing the card interrupt. See MultiFunction Spec for details.

MODEM PRR

7	6	5	4	3	2	1	0
0	0	CReady	0	0	0	RReady	0

Figure 7-9 Modem PRR

Bit 5 CReady: This bit is set to "1" whenever the RReady bit changes state. Writing this bit only has effect when a "1" is simultaneously written to the RReady bit.

Bit 1 RReady: This bit reflects the state of the modem's READY line. Writing this bit does not affect its state, although writing a "1" allows the writing of the CReady bit.

MODEM ESR

7	6	5	4	3	2	1	0
0	0	0	Req Attn	0	0	0	Req Attn En

Figure 7-10 Modem ESR

Bit 4 Req Attn: This bit is set when the STSCTH line is active. This bit is latched as a one. The host writing a one to this bit will reset it to zero. Writing a zero to this bit has no effect.

Bit 0 Req Attn En: when set, the MODEM CSR's Changed bit will be set when the Req Attn bit is set. If the MODEM CSR's SigChange bit is also set, this will generate a STSCHG interrupt. When Req Attn En is cleared, this feature is disabled.

MODEM IOBASE 0

When the MODEM COR bit 1 is set, only I/O accesses in the 8-32 bytes above and including MODEM IOBASE[11:3] will be answered. Otherwise, this register is ignored.

7	6	5	4	3	2	1	0
IOBASE7	IOBASE6	IOBASE5	IOBASE4	IOBASE3	0	0	0

Figure 7-11 Modem IOBASE 0

MODEM IOBASE 1

7	6	5	4	3	2	1	0
0	0	0	0	IOBASE11	IOBASE10	IOBASE9	IOBASE8

Figure 7-12 Modem IOBASE 1

MODEM IOLIMIT

This register controls the size of the modem I/O window. This is an 8 bit register with only two writable bits. Each "1" bit means an address line that is not decoded. Therefore, if the entire register holds 00000111, the window is 8 bytes wide. If the register holds 00011111, the window is 32 bytes.

7	6	5	4	3	2	1	0
0	0	0	IOLIMIT4	IOLIMIT3	1	1	1

Figure 7-13 Modem IOLIMIT

Multi-function Configuration Register

The Multi-function Configuration Register exists in the I/O space of the LAN ASIC. It is loaded upon startup from the serial EEPROM. The secondary function must be connected to the ASIC chip in such a way that a random configuration in this register causes no damage to the card. (i.e. in a design that expects to configure the OE line as a Ready input, a series resistor should be used to prevent contention if this register has not been programmed.) Also, the driver should not activate the modem function without verifying a valid value in this registers, to avoid possible contention. (i.e. if the ModemMultiFunction bit was accidentally set, the modem would answer all cycles sent to the card, almost definitely crashing the host.

Bit	Signal Name	Description
0	MoEarlyWait	The earlyModemWait signal is similar to the earlyReady bit in the LAN chip now. It causes WAIT to be asynchronously asserted at the beginning of a modem cycle (memory or I/O). This assertion only lasts 50-100ns at which point the modem
1	MoBusAB	When clear, the IORD and IOWR lines stay normal. When set, the IORD line becomes DS (Data Strobe - Active low) and the IOWR line becomes R/W (Write Strobe - Active low).
2	MoClkEnable	When set, 1 MHz is output on the MO_CLK line. Otherwise, the line is driven high.
3	MoIreqPolarity	This bit controls the polarity of the interrupt input from the modem. This doesn't invert the input pin, because the pin may also carry Ready/Busy. *1 = active low, 0 = active high.*
4	MoReadyPolarity	This bit controls the polarity of the modem Ready/Busy signal. (as above, doesn't invert pin) 0 = active low, 1 = active high.
5	MoStatusPolarity	This bit controls the polarity of the MO_STSCHG signal, whether it is MO_STSCHG or RING_INDICATE. 0 = active low, 1 = active high.
6	MoWaitPolarity	This bit controls the polarity of the WAIT signal. 0 = active low, 1 = active high.
7	MoWEPolarity	This bit controls the polarity of the WE output. 0 = active low, 1 = active high. This is probably used with MoPowerWE only.
8	MoWEout	When this bit is zero, bit 2 of the MODEM CSR is routed out on the WE line. Use with MoWEPolarity if necessary.
9	MoOEout	When this bit is zero, the OE line becomes the modem's Ready input. In this case, the modem INT input is for interrupts only.
10	MoPCMCIA	When set, PCMCIA cycles are sent to the modem. See next section for details.

11	MoMultiFunction	When set, MultiFunction PCMCIA cycles (IOBASE, IOLIMIT) are also sent out to the modem. I/O cycles will not be qualified based on internal Base/Limit registers.
12	MoStatus	When this bit is set, it is assumed that the external function is providing a CSR, PRR, and ESR. When this bit is set, the MO_STSCHG line is passed through, else the MO_STSCHG line acts as RING.
13	MoUnknown	When set, accesses to unknown attribute locations are sent to the modem. This is to support future chipsets which may implement registers that are currently undefined.
14	MoAllAccess	When set, accesses to all attribute memory is unconditionally passed to the modem chip. This is a debugging mode.
15	MoHALT	When set, the MO_HALT line is asserted. This is a debugging mode.

Changes for 3C589D

SRAM Diagnostic Register (Window 2/Port 06h)

- Bit 3: BIST (R/W)
- Bit 2: BFC (R/W)
- Bit 1: BF (RO)
- Bit 0: BC (RO)

Attribute and Common Memory Access

When the "MoPCMCIA" bit is clear, no attribute accesses go out to the modem and all PCMCIA attribute registers are read from ASIC. The rest of this section assumes that this bit is set. Access to common memory (~REG = 1) are always sent to the modem.

WRITE MEMORY

If the debug bit MoAllAccess is set, all writes are sent to the second function. Assuming this bit is not set: Writes to the COR and CSR are sent to the second function. If MoStatus is set, writes to the PRR and ESR are also sent out. If MoMultiFunction is set, writes to the IOBASE and IOLIMIT registers are also sent out. If MoUnknown is set, writes to unknown registers are also sent out.

READ MEMORY

If the debug bit MoAllAccess is set, all reads are sent to the second function. Assuming this bit is not set: Reads to the COR come from ASIC, not the external function. If MoStatus is set, reads to the CSR, PRR and ESR come from the external function. If MoMultiFunction is set, reads to the IOBASE and IOLIMIT registers come from the external function. If MoUnknown is set, reads to unknown registers are also sent out.

I/O CYCLES When only one function is activated, and the EnableBaseLimit (COR bit 1) is not set, all I/O cycles are answered by that function, without address comparison. If the EnableBaseLimit register bit is set, address comparison with the Base and Limit registers is required. When both functions are activated, the functions must have the Base and Limit registers enabled in order to receive cycles.

If the MoMultiFunction bit is set, it is assumed that the external function has its own Base and Limit registers, so all I/O cycles are passed to the external function without qualification, unless the address comparison determines that the I/O cycles will be answered by the LAN, in which case the cycle is masked from the modem.

Implementation Notes

- When the Ready is coming in from the IREQ line (MoReadyOE = 0), we cannot know the "ready" state of the modem after it is turned on (COR bit 0 = 1). In this case, the aModemReady line is forced high.
- The chip must never generate IOIS16 for the modem. The IOIS8 bit is to tell the chip that the host is 8-bit, not to tell the card to perform 8-bit transfers. A 16-bit host will clear the IOIS8 bit whatever the power-on default is. If multi-function is required to support a 16 bit modem solution, that support will need to exist off chip.
- A bit has been added to the LAN COR to enable the Status Change Interrupt mode.
- The Modem CSR Audio bit should be routed to the COLL pin whevener the modem is active.

Window 0 Configuration Registers

The Window 0 configuration registers control the configuration of the PC Card and provides access to the PC Card's EEPROM.

PCMCIA (3C589, 3C589B, 3C589D)

The Window 0 register set correctly responds to both byte and word I/O cycles. Also, the IRQ drivers are disabled while Window 0 is selected. The following offsets are from the I/O base address.

Manufacturer ID Register (Read Only - Offset 0)

This is the encoded form of 3Com's registered EISA manufacturer code "TCM." The manufacturer code is stored in a byte-swapped format. Refer to Figure 7-14.

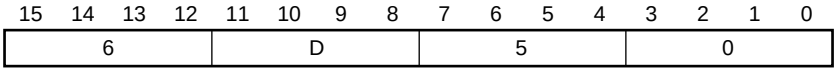


Figure 7-14 Manufacturer ID Register

Product ID Register (3C589) (Read Only - Offset 2)

The automatic configuration logic loads this register from the EEPROM, offset 0x03 (except PCMCIA). The product number is stored in a byte-swapped format. Refer to Figure 7-15.

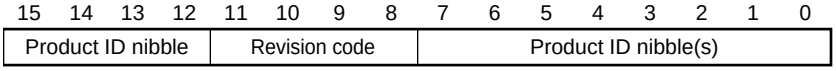


Figure 7-15 Product ID Register

Product ID: The Product ID is the binary-coded decimal 3Com part number for this board. When this changes, a new EISA configuration file will be provided. The product ID is formed by the concatenation of bits 7–0 and bits 15–12, where bits 7–4 form the highest nibble and bits 15–12 form the lowest nibble.

Revision Code: The Revision Code is a 4-bit PC Card revision code. Changes to this field do not require a new version of the EISA configuration file.

PCMCIA (3C589)

This register is not loaded automatically. The register need not be loaded in order for the card to function properly. Have the driver load a Product ID value, if desired, from the EEPROM, offset 3.

Configuration Control Register (3C589) (Read/Write - Offset 4)

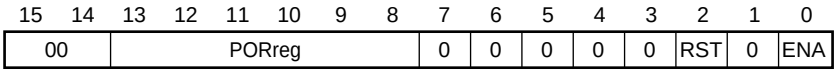


Figure 7-16 Configuration Control Register (3C589)

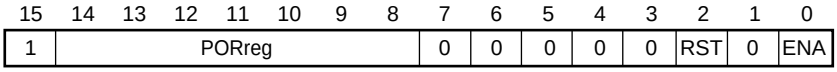


Figure 7-17 Configuration Control Register (3C589B)

PORreg: Power-on reset (POR) jumper register. PORreg is read only. The bit descriptions are as follows:

Bit 14 0 = PCMCIA (3C589B) bus interface.

Bit 13 0 = No AUI connector is available.
1 = AUI connector is available.

Bit 12 0 = No on-board 10BASE2 transceiver is available.
1 = On-board 10BASE2 transceiver is available.

Bits 11–10 00 = Reserved.
01 = Reserved.
10 = Reserved.
11 = Normal operation mode.

Bit 9 0 = No on-board 10BASE-T transceiver is available.
1 = On-board 10BASE-T transceiver is available.

Bit 8 For 3C589:
0 = Use external encoder/decoder.
1 = Use internal encoder/decoder.

For 3C589B:
0 = Use external VCO.
1 = Use internal (on-chip) VCO.

Bit 2 RST (Reset PC Card)
0 = Normal operation.
1 = Reset PC Card to same state as Power-on reset.

Bit 0 ENA (Enable PC Card)
0 = PC Card disabled. Disables IRQ drivers. In EISA mode, it also disables boot PROM address decoding.

In a PCMCIA PC Card, this bit is ignored.

Changes for 3C589D

Configuration Control (Window 0/Port 04h):

- Bit 0: enable - readable/writable but ignored

Address Configuration Register (3C589) (Read/Write - Offset 6)

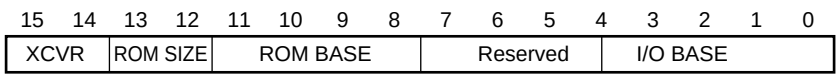


Figure 7-18 Address Configuration Register (3C589)

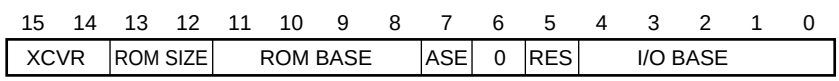


Figure 7-19 Address Configuration Register (3C589B)

Bit 7 contains software configuration information only; that is, it does not control PC Card hardware. This bit is typically used to allow the user to specify to the driver how the hardware is to be configured.

The automatic configuration logic loads this register from the EEPROM, offset 0x08 (except PCMCIA, in which the driver must load the register). Refer to Figure 7-18 and Figure 7-19.

XCVR: **Transceiver Type Select**

00 = Twisted-pair (10BASE-T) transceiver enabled. The software driver must enable Link Beat and Jabber to start the transceiver (refer to Media Type and Status diagnostic register).

01 = AUI port enabled. Using external transceiver.

10 = Reserved: undefined.

11 = BNC (10BASE2) transceiver enabled. The software driver must issue a Start Internal Transceiver command to the Command register to start the DC-DC converter.

PCMCIA (3C589B)

Bit 7 AUTO SELECT. If set, the driver should ignore the XCVR bits and instead auto-select the connector when the driver initializes the PC Card. The connectors available can be determined by examining the Configuration Control register. If AUTO SELECT is clear, the connector indicated in XCVR should be used.

Bit 6 Must be zero.

Bit 5 Reserved.

Bits 4–0 I/O Base Address.
0–30d (0h-1Eh) = Select ISA mode slot-specific I/O address decode.
I/O Base address = Value (in hexadecimal) * 10h + 200h.

31d (1Fh) = Select EISA mode slot-specific I/O address.
I/O Base address = x000 (x = slot number).
Window 0 is also always visible at xC80.

PCMCIA (3C589, 3C589B)

The ROM Size, ROM Base, and I/O Base fields must be set to 0h. This register is not loaded from the EEPROM automatically. This I/O Base is only used internally to the card. If the I/O Base field is not set to 0h, the card will not respond to the I/O. When the I/O Base is set to 0h, the card will respond to I/O accesses on the PCMCIA bus in a modulo 16 manner.

PCMCIA-compliant computers only allow I/O access to the PCMCIA card for which the access is intended. The three most significant nibbles of the address (used to select the card) are superfluous once the computer selects the card. The remaining, least significant, nibble is used to decode the PCMCIA card's 16 ASIC registers.

Changes for 3C589D Address Configuration (Window 0/Port 06h)

- Bits 13-8: romSize, romBase - readable/writable but ignored
- Bits 6: autoSwitch - readable/writable but ignored
- Bits 4-0: ioBase - readable/writable but ignored

Changes for 3C589D

RomControl Register (Window 3/Port 05h) (R/W):

- Bits 1-0: readable/writable but ignored

Resource Configuration Register (3C589, 3C589B) (Read/Write - Offset 8)

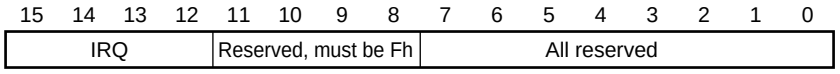


Figure 7-20 Resource Configuration Register (3C589, 3C589B)

The driver loads this register from the EEPROM, offset 0x09. Refer to Figure 7-20 and Figure 7-21.

IRQ: **Interrupt Request select (values given in decimal).**
{3,5,7,9,10,11,12,15} = Enable corresponding IRQ line driver.
{0,1,2,4,6,8,13,14} = Disable all IRQ line drivers.

PCMCIA (3C589, 3C589B)

This register is not loaded automatically. The driver software must set the IRQ field to 3 (3h). Any other value will disable the IRQ line drivers.

Changes for 3C589D

Resource Configuration (Window 0/Port 04h):

- Bits 15-12: irq - readable/writable but ignored

EEPROM Command Register (Read/Write - Offset A)

IPCMCIA (3C589, 3C589B)

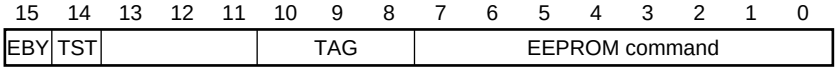


Figure 7-21 EEPROM Command Register (3C589, 3C589B)

TAG: Tag Register (read only). Set by the ID Sequence state machine.

All

EBY: EEPROM Busy status (read only). Refer to Figure 7-23 and Figure 7-24.

0 = EEPROM not busy.

1 = EEPROM busy. I/O writes to the EEPROM command are disabled.

TST: Test Mode (read only). Set at reset time if the EEPROM data input pin is pulled low with external resistor to GND. If set, disables boot PROM. Refer to the section "Test Mode and Bad Configuration Recovery" later in this chapter.

EEPROM Command: Commands written here are shifted out to the on-board EEPROM. There is a 2-bit op code field and a 6-bit address field. For all Erase and Write commands, the hardware times the 10-ms 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 EEPROM Busy status bit to go off before writing the next command.

Table 7-1 EEPROM Commands

Command	OP Code	Address	Data	Exe Time
Read Register	10	a(5:0)	Yes	162 µs
Write Register	01	a(5:0)	Yes	11 ms
Erase Register	11	a(5:0)	No	11 ms
Erase/Write Enable	00	11xxxx	No	60 µs
Erase/Write Disable	00	00xxxx	No	60 µs
Erase All Registers	00	10xxxx	No	11 ms
Write All Registers	00	01xxxx	Yes	11 ms

The Erase commands write all ones into the EEPROM. The Write commands write only zeros. To write data into an EEPROM word, you must issue an Erase/Write Enable (EWEN) command, an Erase command, and an EWEN command; load the data into the EEPROM Data register; and issue a Write command. Remember that you must verify that the EEPROM Busy bit (EEPROM Command register bit 15) is off (zero) before writing a command to the EEPROM Command register or data to the EEPROM Data register.

Changes for 3C589D

AltEepromCommand Register(Window 2/Port 0Ah):

Bit 15 eeBusy(Read Only)

Bit 14 forceConfig(Read Only)

Bit 13 eeSize(Read Only) - 512 bytes(1), 2K bytes(0)

Bit 12 Reserved

Bit 11-0 eeCommand

Table 7-2 ALT EEPROM Commands

	OP Code	Addr	Data	Exe Time
Read Register	10	a(7:0)	Yes	162 µs
Write Register	01	a(7:0)	Yes	11 ms
Erase Register	11	a(7:0)	No	60 µs
Erase/Write Enable	00	11xxxxxx	No	60 µs

EEPROM Data Register (Read/Write - Offset C)

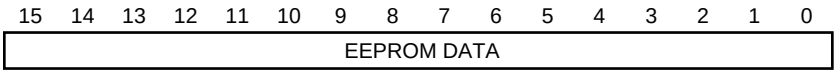


Figure 7-22 EEPROM Data Register

Write data must be written to the EEPROM Data register before the Write command is given to the EEPROM. Read data can be read here after the EEPROM Busy status bit goes off. Refer to Figure 7-22.

Command Register (Read/Write - Offset E)

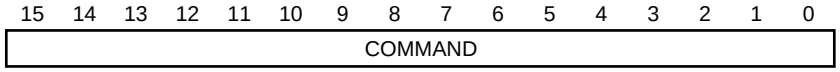


Figure 7-23 Command Register

The only valid commands in Window 0 are Window commands to select another window. Refer to Figure 7-23.

Window 3 Configuration Registers

PCMCIA (3C589B)

Function: Additional setup information.

Location: Window 3/Port 00h

Type: Read/write

Size: 16/32 bits

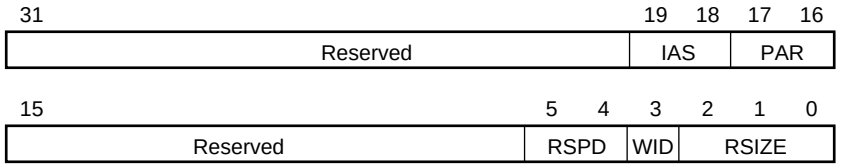


Figure 7-24 Internal Configuration Register

All 32 bits of this register will be loaded from the EEPROM, words 12h and 13h, at startup. Both words of this register can be accessed as 16-bit values. Refer to Figure 7-24.

Bits 31–20 Reserved.

Bits 19–18 ISA ACTIVATION SELECT (3C509B only).

Bits 17–16 RAM PARTITION. Specifies how the external SRAM should be split up into TX FIFO/RX FIFO storage. Defined as the ratio of transmit RAM to receive RAM. Possibilities are (TX:RX):

00b = 3:5 (Only legal if RAM Size = 000b default power-up/reset)
01b = 1:3 (Only legal if RAM Size = 000b)
10b = 1:1
11b = Reserved

Example: RAM WIDTH = byte-wide RAM, RAM SIZE = 8 K and RAM Partition = 1:3. This would imply an 8 KB SRAM (8 Kbytes rather than words), with the TX FIFO using 2 KB, and the RX FIFO 6 KB.

This parameter is expected to be tuned for the PC Card's environment. An 8 KB PC Card in a server that is not having problems with receive overruns should probably set the ratio to 1:1 (4 KB:4 KB) in order to improve transmit performance. An 8 KB Windows client might be best served by the 1:3 (2 KB:6 KB) setting to maximize the size of the RX FIFO to absorb latency without dropped packets.

This parameter can only be changed with both FIFOs in reset state (that is, no input or output since they were reset). The normal sequence would be to issue TX Reset and RX Reset commands just before modifying this parameter

Bits 15–6 Reserved.

Bits 5–4 RAM SPEED. Specifies the number of 20 MHz clocks required for the external SRAM being used on the card. Values are as follows:

00b = 2 clocks (default at power-up/reset). Must use for 3C509B and 3C589B.
01b = 1 clock
10b = Reserved
11b = Reserved

Bit 3 RAM WIDTH. Specifies whether the external RAM is 8 or 16 bits wide. Also affects the interpretation of the RAM SIZE field. Values are as follows:

0b = byte-wide RAM (RAM SIZE in bytes)
1b = word-wide RAM (RAM SIZE in words) (not supported)

This bit is hard-wired and is *not* loaded from the EEPROM for 3C509B (though the EEPROM bit should still be set correctly). The ISA/PCMCIA (3C509B, 3C589B) ASIC will return 0h.

Bits 2–0 RAM SIZE. Specifies the size of the external SRAM used for TX FIFO/RX FIFO storage. The size is either bytes or words depending upon the value of RAM WIDTH. Possibilities are:

000b = 8 K (default at power-up/reset)
010b = 32 K
001b,011b – 111b = reserved

Changes for 3C589D

Internal Configuration Register (Window 3/Port 00h):

Bit 9 (Read/Write) - previously reserved and read as 0.

Used as enable for wakeup circuit; power up or reset 0, disabling the circuit; software can set the bit to allow read/write of attribute memory space while the chip is in Auto Power Down mode. The chip will fall back to sleep at the end of the read/write.

Bit 8 (Read/Write) - previously reserved and read as 0.

Used as PMmode(power management mode bit); power up or reset to 0, meaning conservative power management; software can set to 1 for more aggressive power management.

Bit 5-0 ramSpeed, ramWidth, ramSize - readable/writable but ignored.

EEPROM Data Structure

"X" represents a value that may vary from adapter to adapter.

Offset (Hex)	Field Name	Default (Hex)
00	3Com Node Address (word 0)	XXXX
01	3Com Node Address (word 1)	XXXX
02	3Com Node Address (word 2)	XXXX
03	3C589 Product ID	3C589-TP and 3C589-COMBO: 9058 3C589B-TP and 3C589B-COMBO:9058
04	Manufacturing Data (date)	XXXX
05	Manufacturing Data	XXXX
06	Manufacturing Data	XXXX
07	Manufacturer ID	6D50
08	Address Configuration	XXXX
09	Resource Configuration	XXXX
0A	OEM Node Address (word 0)	XXXX
0B	OEM Node Address (word 1)	XXXX
0C	OEM Node Address (word 2)	XXXX
0D	Software Information	XXXX
0E	Compatibility Word	XXXX
0F	Checksum	XXXX

Offset (Hex)	Field Name	Default (Hex)
PCMCIA (3C589)		
10 to 3F	Network Management Data	XXXX
10	Capabilities Word	2082 for PCMCIA
11	Reserved	0000
12	Internal Configuration Word 0	0000 for PCMCIA
13	Internal Configuration Word 1	0000 for PCMCIA
14	Secondary Software Information	XXXX
15–16	Reserved	0000
17	Secondary Checksums	XXXX
18–3F	Plug and Play Data	(See the section "EEPROM Data Structure, Offsets 18h–3Fh" later in this chapter.)

The manufacturing date format is shown in Figure 7-25.

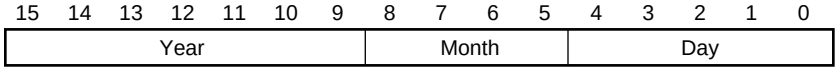


Figure 7-25 Manufacturing Date Format

- Year is 0 through 99 and represents the last two digits of the current year.
- Month is 1 through 12.
- Day is 1 through 31.

EEPROM Data Structure, Offset 0Dh (Software Information)

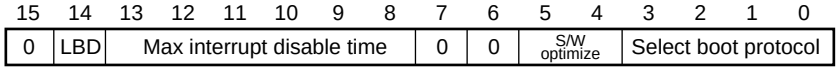


Figure 7-26 EEPROM Data Structure (Software Information)

Bit 14 Link Beat Disable. Based on this bit, the driver should set or reset bit 7 in the Media Type and Status diagnostic register (Window 4/Port 0A). If set to 0 (Enable), the driver should set bit 7 of the Media Type and Status register to 1. If set to 1 (Disable), the driver should set bit 7 to 0. Refer to Figure 7-26.

The 3Com Configuration and Diagnostic Program by default sets this bit to 0 (Enable), according to the 10BASE-T standard. This bit can be set to Disable, however, using the configuration file option.

Link Beat must be disabled to run the transceiver in ENDEC loopback mode.

Bits 13–8 Maximum interrupt disable time. This value, plus one, is the maximum number of 25 microsecond time units that the driver is allowed to disable interrupts (when, for instance, the driver is copying data to or from the adapter FIFO).

For example, if a 9600 baud modem was installed, this value will be set to 19; therefore, the driver should not disable interrupts for longer than 500 microseconds.

Bits 5–4 Software optimization switches. This value reflects the user's system environment, as shown below:

00 = Reserved
01 = DOS client
10 = Windows client
11 = Server

This information can be used by the driver in any desired manner; for example, the driver could choose to allocate more packet buffer space when running in a server environment.

EEPROM Data Structure, Offset 0Eh (Compatibility Word)

The EEPROM data structure, offset 0Eh (Compatibility Word) is shown in Figure 7-27.

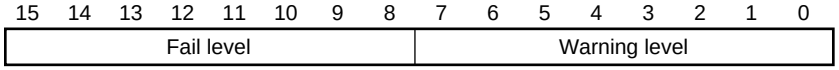


Figure 7-27 EEPROM Data Structure (Compatibility Word)

Bits 15–8 Fail Level. Indicates the hardware revision level as it pertains to software compatibility. For 3C589, this value is zero. For 3C589B, this value is one. If a change is made in the hardware that makes it incompatible with existing drivers, this value will increase. The driver can use this value to compare with its internal level, and if the hardware level is higher, display an error to the user and fail to install itself.

Bits 7–0 Warning Level. Indicates hardware revision level as it pertains to feature or performance enhancements. For 3C589, this value is zero. For 3C589B, this value is one. If a change is made in the hardware that increases features or performance but the hardware is still compatible with existing drivers, this value will increase. The driver can use this value to compare with its internal level, and if the enhancement level is higher, the driver could display a message informing the user to contact the driver vendor to obtain a newer driver.

EEPROM Data Structure, Offset 0Fh (Checksum)

The checksum in the EEPROM is computed as follows:

High byte: Exclusive OR of both bytes of all EEPROM words from offset 00 to offset 0E, inclusive, *except* 08, 09, and 0Dh.

Low byte: Exclusive OR of both bytes of EEPROM words from offsets 08, 09, and 0Dh.

The high byte will normally not change over the lifetime of the PC Card, whereas the low byte will change if the user changes anything using the Configuration and Diagnostic Program.

EEPROM Data Structure, Offset 10h (Capabilities Word)

ISA/PCMCIA (3C589B)

The EEPROM data structure, offset 10h (Capabilities Word for ISA and PCMCIA (3C589B), specifies the capabilities of this PC Card:

Bit 0 Specifies whether the PC Card supports Plug and Play. Set for PCMCIA (3C589B) only.

Bit 1 Always one for 3C589B.

Bits 2–6 Always zero for 3C589B.

Bit 7 CRC PASS THRU. Specifies whether the PC Card supports CRC pass-through or not.

Bits 8–9 Always zero for 3C589B.

Bits 10–12 Reserved, read as zero.

Bit 13 POWER MANAGEMENT. Always set for 3C589B. Implies the power-management commands (Power Up, Power Down Full, and Power Auto) are supported.

Bits 14–15 Reserved, read as zero.

EEPROM Data Structure, Offset 14h (Secondary Software Information)

ISA/PCMCIA (3C589B)

The EEPROM data structure, offset 14h (Secondary Software Information for PCMCIA [3C589B]), is shown in Figure 7-28.

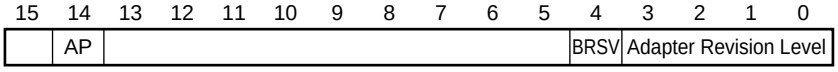


Figure 7-28 Secondary Software Information

Bit 14 00 = Auto Power Enable.
01 = Auto Power Disable.

Bit 4 Boot ROM Size Valid. Indicates that although the Boot ROM Base Address field in the Address Configuration word is zero, there is a boot ROM installed on the PC Card and the Boot ROM Size field is valid. This bit is necessary in the case of a Plug and Play BIOS where the Boot ROM Base Address is filled in by the BIOS, but it should be zero before that to avoid potential conflicts.

Bits 3–0 PC Card Revision Level.

00 = 3C589.
01 = 3C589B.

EEPROM Data Structure, Offset 17h (Secondary Checksums)

ISA/PCMCIA (3C589B)

There are two secondary checksum bytes. The high byte is the vital data checksum; the low byte is the configurable data checksum. The vital data checksum covers words 10h, 11h, and 12h as well as words 20h through 3Fh. The configurable data checksum covers words 13h through 16h. The checksums are the byte XOR (exclusive OR) of the data.

EEPROM Data Structure, Offsets 18h–3Fh

All data stored in these locations is defined and structured according to the ISA Plug and Play Resource Data specifications, and is transferred to the Resource Data register when that register has been selected for reading. (For information on the Resource Data register, refer to the ISA Plug and Play Resource Data specification.)

The contents of the first nine bytes are fixed, and are transferred to the Serial Isolation register according to the Plug and Play Isolation Protocol. These values are listed below:

Offset	Value	Definition
18	6D50	Manufacturer code
19	509X	Product ID
1A	XXXX	Bytes 0 and 1 of node address
1C	8CXX	Fixed value/byte 2 of node address
1A	XX	Checksum

Subsequent items include the Plug and Play and vendor version numbers; an identifier string, logical device ID, IRQ, I/O port, and compatible device descriptor; an optional memory descriptor if a boot ROM is enabled; and appropriate checksums. This data is modified, when required, by an PC Card diagnostic/ configuration program. See the ISA Plug and Play specification for information on decoding this information.

3C589D EEPROM Contents

EEPROM Data Format

Word	Description	Default
00_h	3Com NodeAddress Word 0	0060_h
01_h	3Com NodeAddress Word 1	$97xx_h$
02_h	3Com NodeAddress Word 2	$xxxx_h$
03_h	Product ID	9058_h
04_h	Manufacturing Data - Date	$xxxx_h$
05_h	Manufacturing Data - Division	0036_h
06_h	Manufacturing Data - Product Code	4748_h ‘G’ ‘H’
07_h	Manufacturing ID	$6D50_h$

EEPROM Data Format

Word	Description	Default
08_h	Address Configuration	0090_h for TP, $C090_h$ for coax/combo
09_h	Resource Configuration	$A000_h$
$0A_h$	OEM Node Address Word 0	0060_h
$0B_h$	OEM Node Address Word 1	$97xx_h$
$0C_h$	OEM Node Address Word 2	$xxxx_h$
$0D_h$	Software Information	1320_h
$0E_h$	Compatibility Word	0002_h (0001_h)
$0F_h$	Checksums	$xxxx_h$
10_h	Capabilities	2082_h for PCMCIA
11_h	Reserved	0000_h
12_h	Internal Configuration Word 0	0000_h for ISA/PCMCIA
13_h	Internal Configuration Word 1	0000_h for ISA/PCMCIA
14_h	Adapter Major Revision Level	$C005_h$ (0003_h)
15_h-16_h	Reserved	0000_h
17_h	Secondary Checksums	$xxxx_h$
18_h-$1E_h$	Reserved	0000_h
$1F_h$	Multi-function Configuration Register	1018_h for 3C589D PC Card, $171C_h$ 3C562 PC Card
20_h-$3F_h$	Reserved	0000_h

Notes:
1. The shaded areas must be modified accordingly by Manufacturing to match the Ethernet address on the cover.
2. The values in parentheses are for 3C589C
3. Bits 15:14 of Word 14 are used for Power Management (11=Aggressive Mode)

Data Field Details

NodeAddress

The xx's here represent the bytes of the Node Address that vary from one adapter to another and are allocated from 3Com's reserved block of addresses.

Product ID

This value is loaded from EEProm word 03_h and represents the 3Com-assigned ProductId for this PC Card. For V2 PC Cards this encodes the 3Com *ProductNumber*(509) and a *RevisionCode* used to differentiate between PC Cards. The whole thing is stored in a byte-swapped format as per the EISA spec. The entire field is used to determine what configuration file to use in EISA, including the RevisionCode.

Manufacturing Data - Date

The format is

Bits 15:9 encode the last two digits of the current year(0 thru 99),

Bits 8:5 encode the month(1 thru 12) and

Bits 4:0 encode the day (1 thru 31).

Manufacturing Data - Division

This is the manufacturing division code from the bar code label.

Manufacturing Data - Product Code

This is the manufacturing product code (including hardware revision level). The code is two alpha/numeric ASCII characters read from the bar code label.

ManufacturerId

This is 3Com's assigned EISA ManufacturerID. It is byte-swapped as required by EISA. The value $506D_h$ is the encoded form of the three-character string assigned to 3Com "TCM".

AddressConfiguration

Loaded into the AddressConfiguration register at power-up/reset.

Bits 15-14: *xcvrSelect* - Specifies which transceiver/connector to use. Possibilities are:

00_b: twisted pair(10BaseT) transceiver/RJ45

01_b: external transceiver/AUI

10_b: Reserved - undefined

11_b: BNC(10base2) transceiver/Coax. The host software must issue an *EnableDCConverter* command to start the DC to DC converter.

Bits 13-8: romSize and romBase are readable/writable but ignored.

Bits 7: *autoSelect*. If set new drivers should ignore the *xcvrSelect* bits and instead auto-select the connector when they load. The connectors available can be determined by examining the `ConfigurationControl` register. If *autoSelect* is clear, the connector indicated in *xcvrSelect* should be used.

Bits 6: *autoSwitch* - readable/writable but ignored.

Bits 5: *reserved*, read-as zero.

Bits 4-0: ioBase - readable/writable but ignored.

ResourceConfiguration

Loaded into the ResourceConfiguration register at power-up/reset.

Bits 15-12: irq-readable/writable but ignored.

Bits 11-7: Reserved.

Bit 6: *earlyReady*. Normal I/O cycles if clear. If set, the PC Card will use alternate timing (early deassertion of IOCHRDY/WAIT-). This may allow the board to work in a non-compliant bus that would otherwise not be supported (for example, an ISA bus running faster than spec). Defaults to 0 on power-up, but normally overwritten from EEProm.

Bits 5-0: Reserved.

OEM NodeAddress

This is the node address the host software should actually program as the StationAddress in Window 2. For 3Com PC Cards this will be the same as the 3Com Node Address (words 0-2). OEMs may choose to reprogram this value for their own address block.

Software Information

This field contains information that the host software should use.

Bit 15: Set to enable Full Duplex feature.

Bit 14: Linkbeat disable bit: 0 means enable linkbeat (if TP) and 1 means don't enable linkbeat. See MediaStatus for details.

Bits 13-8: Specify the maximum amount of time the host software can keep interrupts disabled, in multiples of 25us minus one (ie. 000001_b specifies 50us, 111111_b specifies 1600us).

Bits 7-6: Reserved and set to 0.

Bits 5-4: Specify the environment to optimize for: 01_b for DOS clients, 10_b for multitasking clients, and 11_b for servers.

Bits 3: reserved and set to 0.

Bits 2-0: Reserved for LAN-Works (or others) to indicate which boot protocol (out of several possibly supported by the boot ROM) should be used. Reserved and set to 0.

The default value specifies DOS client, 500us disable, and linkbeat enabled.

Compatibility Word

The low byte is the WLevel and the high byte is the FLevel. Drivers are assigned a CLevel, which for V1 drivers is 00_h and for V2 drivers will be 01_h. When a driver installs it will check whether the FLevel is greater than its CLevel. If so it will refuse to install(not the case here). If the WLevel is greater than its CLevel then it will spit out a warning message suggesting that a newer driver is available and may offer improved performance. Since older drivers will work just fine on the newer PC Cards, this will still be kept at 00.

Checksums

Two checksum bytes. The high byte is the vital data checksum, the low byte is the configurable data checksum. The configurable data checksum covers words 08_h, 09_h and $0D_h$. The vital data check covers the remaining words in 00_h thru $0E_h$. The checksums are the byte xor of data.

Capabilities Word

Specifies the capabilities of this PC Card:

Bits 15-14: reserved, read as zero.

Bit 13: supportsPowerManagement. Always set for V2. Implies the power-management commands(PowerUp, PowerDownFull and PowerAuto) are supported.

Bits 12-10: reserved, read as zero.

Bit 9: *supportsIgnoreLength*. Always zero for V2.

Bit 8: *supportsTxDone*. Always zero for V2.

Bit 7: *supportsCrcPassThru*. Specifies whether the PC Card supports CRC pass-thru or not. Always set for V2.

Bit 6: *supportsFragBusMaster*. Always zero for V2.

Bit 5: *supportsFullBusMaster*. Always zero for V2.

Bit 4: *supports2ndDma*. Always zero for V2.

Bit 3: *supportsSlaveDma*. Always 0 in V2.

Bit 2: *supportsLargePackets*. Always 0 in V2.

Bit 1: *supportsFullDuplex*. Specifies whether the PC Card supports full-duplex 10BaseT operation with the appropriate (bridge-per-port) hub, allowing 20Mbps operation. Always 1 in V2.

Bit 0: *supportsPnP*. Specifies whether the PC Card supports PlugAndPlay. Set for V2 ISA only.

Internal Configuration

All 32 bits of this register will be loaded from the EEProm, words 12 and 13, at startup, allowing for possible extensions for external bus interface chips snooping the bus. Both words of this register can be accessed as 16-bit values.

Bits 31-18: Reserved, read as zeroes.

Bits 17-16: ramPartition. Specifies how the external SRAM should be split up into TxFIFO/RxFIFO storage. Defined as the ratio of transmit RAM to receive RAM. Possibilities are (Tx:Rx).

00_b=3:5(only legal if ramSize=000_b, default power-up/reset)

01_b=1:3(only legal if ramSize=000_b)

10_b=1:1

11_b=reserved

Bits 15-10: Reserved, read as zeroes.

Bit 9: Previously reserved and read as 0. Used as enable for wakeup circuit; power up or reset to 0, disabling the circuit; software can set the bit to allow read/write of attribute memory space while the chip is in Auto Power Down mode. The chip will fall back to sleep at the end of the read/write.

Bit 8: Previously reserved and read as 0. Used as PMmode(Power management mode bit) in ASIC; power up or reset to 0, meaning conservative power management; software can set to 1 for more aggressive power management.

Bit 7-6: Reserved, read as zeroes.

Bit 5-0: *ramSpeed,ramWidth, ramSize* - readable/writable but ignored.

PC Card Major Revision Level

PC Card Major Revision Level- Secondary Software Information

Bit 15:14 are used for Power Management

00_b=Reserved, read as zero.

01_b=Reserved, read as zero.

10_b=Reserved, read as zero.

11_b=Aggresive Power Management Mode

Bit 13:5 reserved, read as zero.

Bit 4: Boot ROM Size Valid. Indicates that although the Boot ROM Base Address field in the Address Configuration word is zero, there is a boot ROM installed on the PC Card and the Boot ROM Size field is valid. This bit is necessary in the case of a Plug and Play BIOS where the Boot ROM Base

Address is filled in by the BIOS, but it should be zero before that to avoid potential conflicts.

Bit 3:0 Adapter Revision Level.

00_b=3C589

01_b=3C589B.

Secondary Checksums

Two secondary checksum bytes. The high byte is the vital data checksum, the low byte is the configurable data checksum. The vital data checksum covers words 10_h, 11_h, and 12_h as well as words 20_h thru $3F_h$. The configurable data checksum covers words 13_h thru $1E_h$. The checksums are the byte xor of the data.

PCMCIA-Specific Data Structures (3C589)

The Card Information Structure (CIS) used by the 3Com PCMCIA (3C589) PC Card is listed below. Refer also to the section "PCMCIA Activation Mechanism" earlier in this chapter.

```
--- TUPLES.ASM -----------------------------------------------------
Tuple data for 3Com PCMCIA Card (3C589)
When this data is written to Attribute memory on the card it will occupy only even byte
addresses, so offsets in this listing should be multiplied by two to get the actual memory
locations on the card.

The checksum value in _csum_val must be computed if data changes.

                        CIS         segment

0000                    _cis_start  label  byte

                        ;Common Memory Device Tuple
0000  01                tpl_dev     db     01     ;tuple id
0001  02                            db     (cmend - $ - 1) ;link
0003  FF                            db     0ffh   ;no size / end
0004                    cmend       label  byte

                        ;Attribute Memory Device
                        Tuple
0004  17                tpl_deva    db     17h    ;tuple id
0005  03                            db     (amend - $ - 1) ;link
0006  43                            db     43h    ;eeprom, 150 ns
0007  02                            db     02     ;8k
0008  FF                            db     0ffh   ;end
0009                    amend       label  byte
0004                    cmend       label  byte

                         ;Manufacturer ID Tuple
0009  20                tpl_manid   db     20h    ;tuple id
```

```
000A  04                              db     (midend - $ - 1) ;link
000B  0101                            dw     0101h  ;3Com ID
000D  0589                            dw     0589h  ;Card type ID: low 3 nibbles family
                                                    ;ID high nibble type/revision
000F                        midend    label  byte

                            ;Function ID Tuple
000F  21                    tpl_funud db     21h    ;tuple ID
0010  02                              db     (fidend - $ - 1) ;link
0011  06                              db     06     ;LAN adapter
0012  00                              db     0      ;no Post init, no ROM
0013                        fidend    label  byte

                            ;Version 1 Tuple
0013  15                    tpl_versl db     15h    ;tuple ID
0014  38                              db     (verend - $ - 1) ;link
0015  04                              db     04     ;major vers
0016  01                              db     01     ;minor vers
0017  33 43 6F 6D 20 43              db     '3Com Corporation',0        ;mfgr name
      6F 72 70 6F 72 61
      74 69 6F 6E 00
0028  33 43 35 38 39 00              db     '3C589',0                   ;product name
002E  54 50 2F 42 4E 43              db     'TP/BNC LAN CardVer.1a',0   ;product info 1
      20 4C 41 4E 20 43
      61 72 64 20 56 65
      72 2E 20 31 61 00
0046  30 30 30 30 30 31              db     '000001'                    ;product info 2
004C  FF                              db     offh   ;end
004D                        verevd    label  byte

                            ;LAN Function Extension Tuple
                                      ;not yet

                            ;Configuration Tuple
004D  1A                    tpl_config db    lah    ;tuple ID
004E  05                              db     (cfgend - $ - 1) ;link
004F  01                              db     01     ;2 bytes of Reg Base Addr,
                                                    ;1 byte of Reg Presence
0050  03                              db     03     ;last config entry index = 3
0051  00                              db     00
0052  20                              db     20h    ;reg base = 2000h (8K boundary)
0053                        verevd    label  byte
0054                        cfgend    label  byte

                            ;Configuration Entry Subtuple
```

```
0054  1B                tpl_cftab1  db     1bh     ;subtuple ID
0055  0E                            db     (ceend1 - $ - 1) ;link
0056  C1                            db     0clh    ;entry #1, default, Intf Cnfg byte
                                                   ;follows
0057  01                            db     01      ;I/O interface
0058  1D                            db     1dh     ;Features: Vcc, Timing, I/O, IRQ
0059  71                            db     71h     ;Vcc Power: Ipdn, Ipk, Iavg, Vnom
005A  55                            db     55h     ;  Vnom = 5 V
005B  1E                            db     1eh     ;  Iavg = 150 mA
005C  26                            db     26h     ;  Ipk  = 200 mA
005D  05                            db     05h     ;  Ipdn = 10 mA
005E  E7                            db     0e7h    ;no Wait, Rdy/Busy x10, no Reserved
005F  26                            db     26h     ;  Rdy/Bsy time 15 x 10 = 150mS
0060  64                            db     64h     ;I/O 8/16-bit, on 16-byte boundary
0061  20                            db     20h     ;IRQ Level mode, mask follows
0062  FF                            db     0ffh    ;  IRQ 0-7 ok
0063  FF                            db     0ffh    ;  IRQ 8-15 ok
0064                    ceendl      label  byte

                        ;Configuration Entry Subtuple
0064  1B                tpl_cftab3  db     1bh     ;subtuple ID
0065  0E                            db     (ceend3 - $ - 1) ;link
0066  83                            db     083h    ;entry #3, Intf Cnfg byte ;follows
0067  01                            db     01      ;I/O interface
0068  1D                            db     1dh     ;Features: Vcc, Timing, I/O, IRQ
0069  71                            db     71h     ;Vcc Power: Ipdn, Ipk, Iavg, Vnom
006A  55                            db     55h     ;  Vnom = 5 V
006B  5E                            db     5eh     ;  Iavg = 550 mA
006C  66                            db     66h     ;  Ipk  = 600 mA
006D  05                            db     05h     ;  Ipdn = 10 mA
006E  E7                            db     0e7h    ;no Wait, Rdy/Busy x10, no Reserved
006F  26                            db     26h     ;  Rdy/Bsy time 15 x 10 = 150mS
0070  64                            db     64h     ;I/O 8/16-bit, on 16-byte boundary
0071  20                            db     20h     ;IRQ Level mode, mask follows
0072  FF                            db     0ffh    ;  IRQ 0-7 ok
0073  FF                            db     0ffh    ;  IRQ 8-15 ok
0074                    ceend3      label  byte

                        ;JEDEC Device Info, Attribute Memory Tuple
0074  19                tpl_jedec_a db     19h     ;tuple ID
0075  03                            db     (jaend3 - $ - 1) ;link
0076  00                            db     0       ;no code
0077  00                            db     0       ;no info
0078  FF                            db     0ffh    ;end
```

```
0079                    jaend       label  byte

                        ;NoLink Tuple
0079  14                tpl_nolink  db     14h    ;tuple ID
007A  00                            db     0      ;link

                        ;Checksum Tuple
007B  10                tpl_chksum  db     10h    ;tuple ID
007C  05                            db     (csend3 - $ - 1) ;link
007D  FF85                          dw     (_cis_start - tpl_chksum)   ;start rel to
                                                                       ;this tuple
007F  0084              cks_size    dw     CKS_LENG
0081  00                            db     0      ;checksum value - leave this zero!
                                                   ;Follow PCMCIA spec, but fix
                                                   ;limitations
0082                    csend       label  byte

                        ;End Tuple
0082  FF                tpl_end     db     0ffh   ;end-of-chain tuple
                                    ;*******************************************

0083  AC                _real_csum  db     0ach   ;value computed for preceding data

= 0084                  CKS_LENG    equ    ($ - _cis_start)            ;length for
                                                                       ;checksum tuple
0088                    CIS         ends
                                    end
```

PCMCIA-Specific Data Structures (3C589B)

The Card Information Structure (CIS) used by the 3Com PCMCIA (3C589B) PC Card is listed below. Refer also to the section "PCMCIA Activation Mechanism" earlier in this chapter.

```
--- TUPLES.ASM -----------------------------------------------------

Tuple data for 3Com PCMCIA Card (3C589B)

When this data is written to Attribute memory on the card it will occupy only even byte
addresses, so offsets in this listing should be multiplied by two to get the actual memory
locations on the card.

Format is designed to be written to card by a C program. The checksum value in _csum_val must be
computed to complete tuples; the C program generates this checksum value before writing to card.

                        CIS         segment

0000                    _cis_start  label  byte

                        ;Common Memory Device Tuple
0000  01                tpl_dev     db     01     ;tuple id
```

```
0001  02                                  db      (cmend - $ - 1) ;link
0003  FF                                  db      0ffh    ;no size / end
0004                        cmend         label   byte

                            ;Attribute Memory Device Tuple
0004  17                     tpl_deva     db      17h     ;tuple id
0005  03                                  db      (amend - $ - 1) ;link
0006  43                                  db      43h     ;eeprom, 150 ns
0007  02                                  db      02      ;8k
0008  FF                                  db      0ffh    ;end
0009                        amend         label   byte


                          ;Manufacturer ID Tuple
0009  20                   tpl_manid     db      20h     ;tuple id
000A  04                                 db      (midend - $ - 1) ;link
000B  0101                               dw      0101h   ;3Com ID
000D  0589                               dw      0589h   ;Card type ID: low 3 nibbles family
                                                         ;ID high nibble type/revision
000F                        midend       label   byte

                          ;Function ID Tuple
000F  21                   tpl_funud     db      21h     ;tuple ID
0010  02                                 db      (fidend - $ - 1) ;link
0011  06                                 db      06      ;LAN adapter
0012  00                                 db      0       ;no Post init, no ROM
0013                        fidend       label   byte

                          ;Version 1 Tuple
0013  15                   tpl_versl     db      15h     ;tuple ID
0014  3E                                 db      (verend - $ - 1) ;link
0015  04                                 db      04      ;major vers
0016  01                                 db      01      ;minor vers
0017  33 43 6F 6D 20 43                  db      '3Com Corporation',0        ;mfgr name
      6F 72 70 6F 72 61
      74 69 6F 6E 00
0028  33 43 35 38 39 00                  db      '3C589',0                   ;product name
002E  54 50 2F 42 4E 43                  db      'TP/BNC LAN Prototype       ;product info 1
      20 4C 4E 20 50 72                          Ver. 2a',0
      6F 74 6F 74 79 70
      65 20 56 65 72 2E
      20 32 61 00
004B  30 30 30 30 30 32 00               db      '000002',0                  ;product info 2
0052  FF                                  db      offh    ;end
0053                        verevd       label   byte
```

```
                        ;LAN Function Extension Tuple
                                    ;later

                        ;Configuration Tuple
0053  1A                tpl_config    db      lah     ;tuple ID
0054  06                              db      (cfgend - $ - 1) ;link
0055  02                              db      02      ;3 bytes of Reg Base Addr,
                                                      ;1 byte of Reg Presence
0056  03                              db      03      ;last config entry index = 3
0057  00                              db      00
0058  00                              db      00
0059  01                              db      01h     ;reg base = 10000h (64K boundary)
005A  03                              db      03      ;Reg 0, 1; COR, STAT
005B                    cfgend        label   byte

                        ;Configuration Entry Subtuple
005B  1B                tpl_cftab1    db      1bh     ;subtuple ID
005C  0E                              db      (ceend1 - $ - 1) ;link
005D  C1                              db      0clh    ;entry #1, default, Intf Cnfg byte
                                                      ;follows
005E  01                              db      01      ;I/O interface
005F  1D                              db      1dh     ;Features: Vcc, Timing, I/O, IRQ
0060  71                              db      71h     ;Vcc Power: Ipdn, Ipk, Iavg, Vnom
0061  55                              db      55h     ;  Vnom = 5 V
0062  1E                              db      1eh     ;  Iavg = 150 mA
0063  26                              db      26h     ;  Ipk  = 200 mA
0064  05                              db      05h     ;  Ipdn = 10 mA
0065  E7                              db      0e7h    ;no Wait, Rdy/Busy x10, no Reserved
0066  26                              db      26h     ;  Rdy/Bsy time 15 x 10 = 150mS
0067  64                              db      64h     ;I/O 8/16-bit, on 16-byte boundary
0068  30                              db      30h     ;IRQ Level mode, mask follows
0069  FF                              db      0ffh    ;  IRQ 0-7 ok
006A  FF                              db      0ffh    ;  IRQ 8-15 ok
006B                    ceendl        label   byte

                        ;Configuration Entry Subtuple
006B  1B                tpl_cftab3    db      1bh     ;subtuple ID
006C  0E                              db      (ceend3 - $ - 1) ;link
006D  83                              db      083h    ;entry #3, Intf Cnfg byte follows
006E  01                              db      01      ;I/O interface
006F  1D                              db      1dh     ;Features: Vcc, Timing, I/O, IRQ
0070  71                              db      71h     ;Vcc Power: Ipdn, Ipk, Iavg, Vnom
0071  55                              db      55h     ;  Vnom = 5 V
0072  5E                              db      5eh     ;  Iavg = 550 mA
```

```
0073  66                              db      66h     ;  Ipk  = 600 mA
0074  05                              db      05h     ;  Ipdn = 10 mA
0075  E7                              db      0e7h    ;no Wait, Rdy/Busy x10, no Reserved
0076  26                              db      26h     ;  Rdy/Bsy time 15 x 10 = 150mS
0077  64                              db      64h     ;I/O 8/16-bit, on 16-byte boundary
0078  30                              db      30h     ;IRQ Level mode, mask follows
0079  FF                              db      0ffh    ;  IRQ 0-7 ok
007A  FF                              db      0ffh    ;  IRQ 8-15 ok
007B                        ceend3    label   byte

                            ;JEDEC Device Info, Attribute Memory Tuple
007B  19                    tpl_jedec_a db      19h     ;tuple ID
007C  03                              db      (jaend3 - $ - 1) ;link
007D  00                              db      0       ;no code
007E  00                              db      0       ;no info
007F  FF                              db      0ffh    ;end
0080                        jaend     label   byte

                            ;NoLink Tuple
0080  14                    tpl_nolink  db      14h     ;tuple ID
0081  00                              db      0       ;link

                            ;Checksum Tuple
0082  10                    tpl_chksum  db      10h     ;tuple ID
0083  05                              db      (csend3 - $ - 1) ;link
0084  FF7E                            dw      (_cis_start - tpl_chksum)   ;start rel to
                                                                          ;this tuple
0086  008B                  cks_size    dw      CKS_LENG
0088  00                              db      0       ;checksum value - leave this zero!
                                                      ;PCMCIA spec is bad, see Warning
                                                      ;below
0089                        csend     label   byte

                            ;End Tuple
0089  FF                    tpl_end     db      0ffh    ;end-of-chain tuple
      ;*****************************************************************************
      ;WARNING:
      ; PCMCIA Committee bad definition. The checksum can't be computed if you
      ;include the tuple's checksum byte in the chksum range. To fix this, define
      ;the checksum value in the tuple as equal to zero and add an extra byte
      ;after end of tuples to be the real checksum. Value of this extra byte
      ;should be set to make sum across the checksum range equal zero. The
      ;_real_csum value will be set correctly by the routine that writes CIS.
      ;
      ;Don't move next two lines from their position folowing tpl_end.
      ;*****************************************************************************
```

```
008A  00                  _real_csum    db      0       ;value filled in later by cis-
                                                        ;writing program

=008B                     CKS_LENG      equ     ($ - _cis_start)            ;length for
                                                                            ;checksum tuple
008B  0300                _io_base      dw      0300h   ;Configured io base address to aid
                                                        ;config for driver that directly
                                                        ;writes to Intel PCIC. Should match
                                                        ;io address written in eeprom. Must
                                                        ;follow real csum in Attribute
                                                        ;memory.
008D  008D                _cis_size     dw      9_cis_size - _cis_start)    ;length of CIS
                                                                            ;data to write
008F                      CIS           ends
                                        end
```

8 PCMCIA BUS INTERFACES

Supported Slot Types and Cycle Types

Table 8-1 summarizes the slot types and cycle types supported by the 3C589 PC Cards.

Table 8-1 EtherLink III Adapter Slot and Cycle Types

Bus Interfaces, Slot Types	I/O Cycles Supported	Memory Read Cycles Supported	Other Supported Features
PCMCIA	8-bit (ISA-like) and 16-bit (always asserts –IOIS 16)	8-bit transfers to Attribute Memory*	Only default timing, as specified in the PCMCIA release 2.01, PC Card Specification.

* PCMCIA Configuration registers and Card Information Structure.

PCMCIA Details

Refer to Table 8-2 for PCMCIA pin assignments.

Table 8-2 PCMCIA Pin Assignments (Pins 1 Through 68)

Pin	Signal	I/O Type	Function	Polarity	Notes
1	GND		Ground		
2	D3	I/O	Data bit 3		
3	D4	I/O	Data bit 4		
4	D5	I/O	Data bit 5		
5	D6	I/O	Data bit 6		
6	D7	I/O	Data bit 7		
7	CE1	I	Card enable	AL	3
8	A10	I	Address bit 10		
9	OE	I	Output enable	AL	
10	A11	I	Address bit 11		
11	A9	I	Address bit 9		
12	A8	I	Address bit 8		
13	A13	I	Address bit 13		
14	Not used				
15	WE/PGM	I	Write enable	AL	
16	Not used				
17	Vcc				
18	Not used				
19	Not used				
20	Not used				
21	A12	I	Address bit 12		
22	A7	I	Address bit 7		

Table 8-2 PCMCIA Pin Assignments (Pins 1 Through 68) (continued)

Pin	Signal	I/O Type	Function	Polarity	Notes
23	A6	I	Address bit 6		
24	A5	I	Address bit 5		
25	A4	I	Address bit 4		
26	A3	I	Address bit 3		
27	A2	I	Address bit 2		
28	A1	I	Address bit 1		
29	A0	I	Address bit 0		
30	D0	I/O	Data bit 0		
31	D1	I/O	Data bit 1		
32	D2	I/O	Data bit 2		
33	Not used				
34	GND		Ground		
35	GND		Ground		
36	CD1	O	Card detect	AL	3
37	D11	I/O	Data bit 11		
38	D12	I/O	Data bit 12		
39	D13	I/O	Data bit 13		
40	D14	I/O	Data bit 14		
41	D15	I/O	Data bit 15		
42	CE2	I	Card enable	AL	3
43	Not used				
44	Not used				
45	Not used				
46	Not used				
47	Not used				
48	Not used				
49	Not used				
50	Not used				
51	Vcc				
52	Not used				
53	Not used				
54	Not used				
55	Not used				
56	Not used				
57	RFU		Reserved		
58	RESET	I	Card reset	AH	1,4
59	WAIT	O	Extend bus cycle	AL	1,3
60	Not used				
61	REG	I	Register select and I/O enable	AL	2
62	Not used				
63	Not used				

Table 8-2 PCMCIA Pin Assignments (Pins 1 Through 68) (continued)

Pin	Signal	I/O Type	Function	Polarity	Notes
64	D8	I/O	Data bit 8		
65	D9	I/O	Data bit 9		
66	D10	I/O	Data bit 10		
67	CD2	O	Card detect	AL	3
68	GND		Ground		

Notes:
I/O Type column symbols: I=Input to card; O=Output from card; I/O=Bidirectional
Polarity column symbols: AH=Active High; AL=Active Low

1. Wait and Reset, which are RFU (no connect) in release 1.0 of the PCMCIA PC Card Standard, must be implemented to comply with release 2.0 of the standard.
2. Signals in this I/O and memory card are used differently from the corresponding pins of memory-only PCMCIA cards.
3. Do not connect signal between cards. Do not directly connect (wire-OR or wire-AND) with any signal source in the host.
4. Do not connect Reset between cards unless you want all cards to reset when Vcc is removed from any of the cards.

9 EXTERNAL CONFIGURATION OPTIONS

The 3C589 PC Card's ASIC supports a wide variety of external configuration options. These options are available to support board-level testing and notify the ASIC and the host software of media resource support.

Boundary Scan Configuration

When ResetPinIn is asserted, a boundary scan ring is given access to the ASIC's I/O pins. RData[7:4] acts as the control port to the boundary scan with the following functionality:

Bit	Function	I/O
RData[7]	DataIn	Input
RData[6]	DataOut	Output
RData[5]	BClk	Input
RData[4]	Shift/Load	Input

Changes for 3C589D

Scan Mode is RESET high and IOR & IOW are low.

RESET = 1;

IOR = 0:

IOW = 0;

These I/O pins are available to boundary scan.

Scan chain Inputs:

IOIS16

A[0:9]

A11

A10

REG

INPACK

WAIT

D[7:0]

D[15:8]

CE1

CE2

OE

WE

READY

Scan chain Output:

STSCHG.

Forced Configuration

The Ethernet controller ASIC is always powered-up in this mode. It is configured with an internal default I/O base address of 200h and with the internal 10BASE-T transceiver enabled independently of the availability of this resource. Note that although the Address Configuration is loaded for an I/O base of 200h, the card responds on the PCMCIA bus to any modulo 16 I/O address.

Physical Layer Configuration

RData[5:0] are used to notify the ASIC of its external physical layer resources. As with forced configuration, these pins are sampled on the falling edge of ResetPinIn, and various configurations are strapped via pulldown resistors on the PC board.

These configurations must not be modified by the end user.

Physical layer resources are encoded as follows:

RData	Resources Available
0	Internal encoder/decoder
1	10BASE-T transceiver (through internal interface)
2	See "Physical Layer Test Access" section.
3	See "Physical Layer Test Access" section.
4	10BASE2 transceiver (through AUI and DC-DC converter)
5	15-pin AUI connector

Except for the external encoder/decoder configuration, an internal AUI transceiver is available in all configurations.

The external encoder/decoder configuration is for testing purposes only.

Physical Layer Test Access

RData[3:2] (as sampled at the falling edge of ResetPinIn) allow access to the physical layer for test purposes. The various access ports are windowed to the RData[7:0] bus, and are listed below.

Pin 3	Pin 2	Mode
0	0	Reserved
0	1	Receive Physical Access
1	0	Transmit Physical Access
1	1	Normal EEPROM Access (Default)

Access to the Receive and Transmit physical tests occurs at test points TP22 and TP23. TP16 must also be pulled down (as sampled at the trailing edge of ResetPinIn) to go into this test mode.

10 PC CARD DIFFERENCES

This chapter contains topics that explain differences among the various types of 3C589 PC Cards. The 3C589 PC Cards support the Card Information Structure (CIS), Configuration Option register, and Card Configuration and Status register defined in the PCMCIA specification.

Changes in PCMCIA Adapters

Significant changes have been made in the current (3C589B) version of the EtherLink PC Cards.

This section summarizes the changes inPCMCIA (3C589B) adapters relative to the previous version (3C589). These changes may affect the functionality of the host software.

External RAM

RAM is now external rather than on-chip. The 3C589 had 4 K of on-chip RAM. The 3C589B now uses external SRAM. The size of the SRAM can vary from 8 K to 32 K. The 3C589B adapter use 32 K. The Internal Configuration register was added to support this.

RAM Allocation

The allocation of RAM to transmit and receive FIFOs can be changed. The 3C589 allocated 2 K to the TX FIFO and 2 K to the RX FIFO. 3C509B can now allocate RAM to the TX FIFO and RX FIFO in the following ratios: 2:6, 3:5, and 4:4, scaled by the number of 8 K blocks of memory available. The 3C589B ratio is fixed at 4:4.

PCMCIA Support

PCMCIA (3C589B) support is provided on-chip, rather than through external logic, as with the 3C589. A low-power mode is provided.

Capabilities Word

A Capabilities Word has been added to the EEPROM contents, allowing host software to determine precisely whether individual functions are present or not.

Advanced Power-Management Features

The 3C589B support advanced power-management features. The boards can be almost completely powered-down, and with a 10BASE-T connector, the board can be put to sleep during idle periods between packets.

8-Bit Access Support

The 3C589B support 8-bit accesses, which the 3C589 adapters did not. This will allow them to be used in 8-bit slots or in 8-bit designs (like handhelds), where 3C589 adapters could not be used.

Threshold Values Changed

The default/power-up values of some of the thresholds have changed. Software should treat any value greater than the maximum packet size (1,792 bytes) as disabled for all thresholds.

Power-Up Values Changed

The default TX Free/RX Free values at power-up have changed, and vary with the amount of RAM on the adapter. The software should not rely on the exact empty value. If the software needs this value, it should read TX Free/RX Free at startup after issuing a TX Reset/RX Reset command to guarantee the FIFO is empty.

Extra Bytes Required

The hardware implementations of TX FIFO and RX FIFO require a few extra overhead bytes per packet beyond that needed in previous versions of EtherLink III adapters. The software should not rely on the exact number of bytes registered in TX Free/RX Free for a given packet.

Changes in the PCMCIA Adapters

A few of the changes apply to the ISA and PCMCIA adapters (3C589B). These changes are summarized below.

Postamble

The postamble can no longer be read from the RX FIFO directly. An RX Discard command is needed to advance to the next packet.

The same change to the RX FIFO design made obsolete the RX Status Overrun error, which can no longer occur. The RX FIFO can now contain an arbitrary number of packets.

Revised Reset Commands

The Global Reset, TX Reset, and RX Reset commands are now word-sized and have bit masks associated with them, allowing selective resets of certain modules in hardware only.

Changed RX Bytes Count

The RX Bytes count in the RX Status register now increments by 4 until the packet is complete, helping the software to keep on dword boundaries if it started out that way.

Elimination of Hidden Bytes

There are no longer any hidden bytes in the RX FIFO implementation.

RX Early Settings

Setting RX Early to values less than 8 is the same as setting it to 8 bytes.

RX Discard Command

An RX Discard command issued before the receive packet becomes visible to the host is ignored. Previously an RX Discard of such a packet would still discard it.

Transmit Preamble

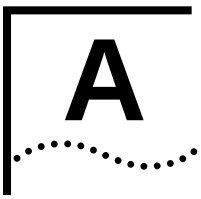
The transmit preamble for the current packet is read as soon as it is written by the host. Therefore it will not cause a decrease in the TX Free value.

TX Start Threshold Command

The TX Start Threshold that can be read in Window 5 now reads the value written by the Set TX Start Threshold command, rather than that value plus 4.

Host Diagnostic Register Removed

The Host Diagnostic register has been removed.



DRIVER ROUTINE FLOWCHARTS

This appendix contains process flowcharts. They help software developers write code that best incorporates the features of the 3C589 adapters in their drivers. These flowcharts help write routines for the three most important functions of an adapter; that is, initialize the adapter, transmit frames of data, and receive frames of data. The flowcharts are as follows:

- Initialize Adapter - This routine initializes the hardware.
- Get Bus Type - This routine identifies the type of bus (PCMCIA).
- Check Slot (PCMIA) - This routine collects specific information about the bus identified in the previous routine.
- Write ID Sequence - This routine writes the ID sequence to the adapter.
- Transmit Frame - This routine sets up the adapter to transmit frames.
- Transmit Output - This routine is called by transmit and interrupt routines.
- Interrupt Service Routines - These are the four types of interrupt routines for transmit and receive frames.
- Timer Interrupt - This routine uses the timer interrupt sent from the computer hardware to adjust driver variables.

Initialize Adapter (PCMCIA)

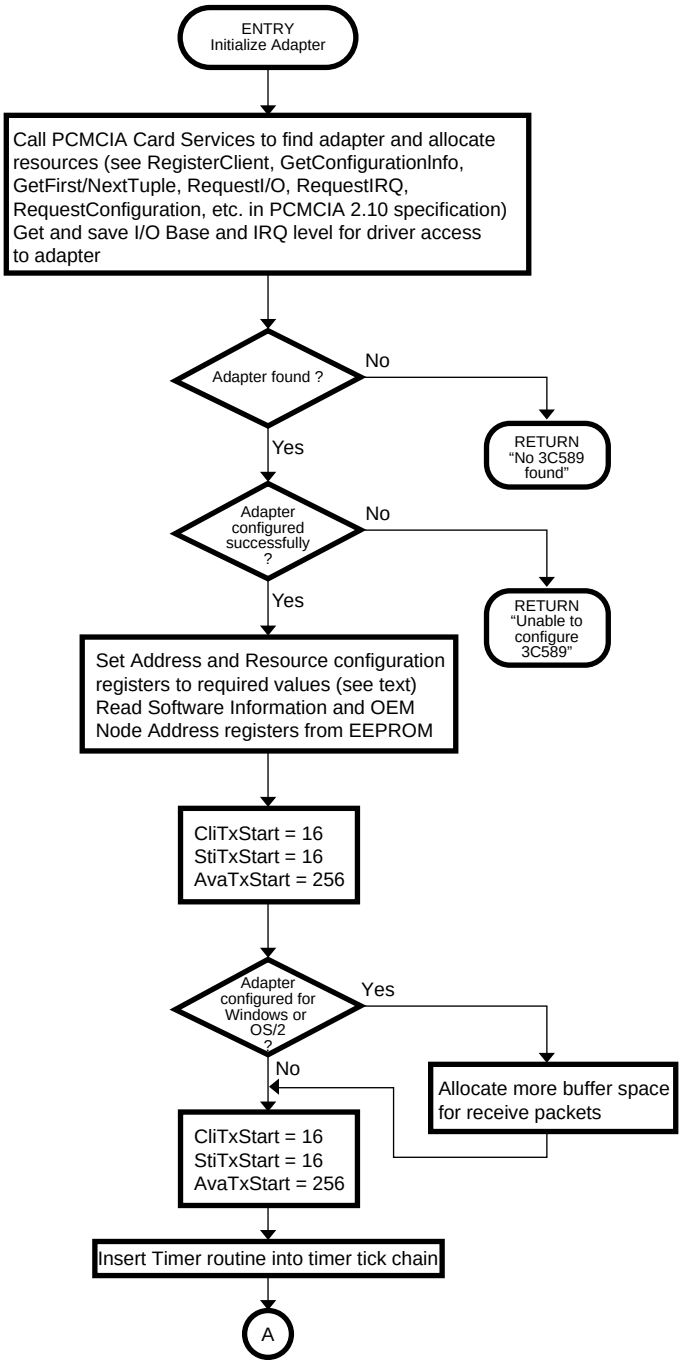


Figure A-1 Initialize Adapter (PCMCIA)

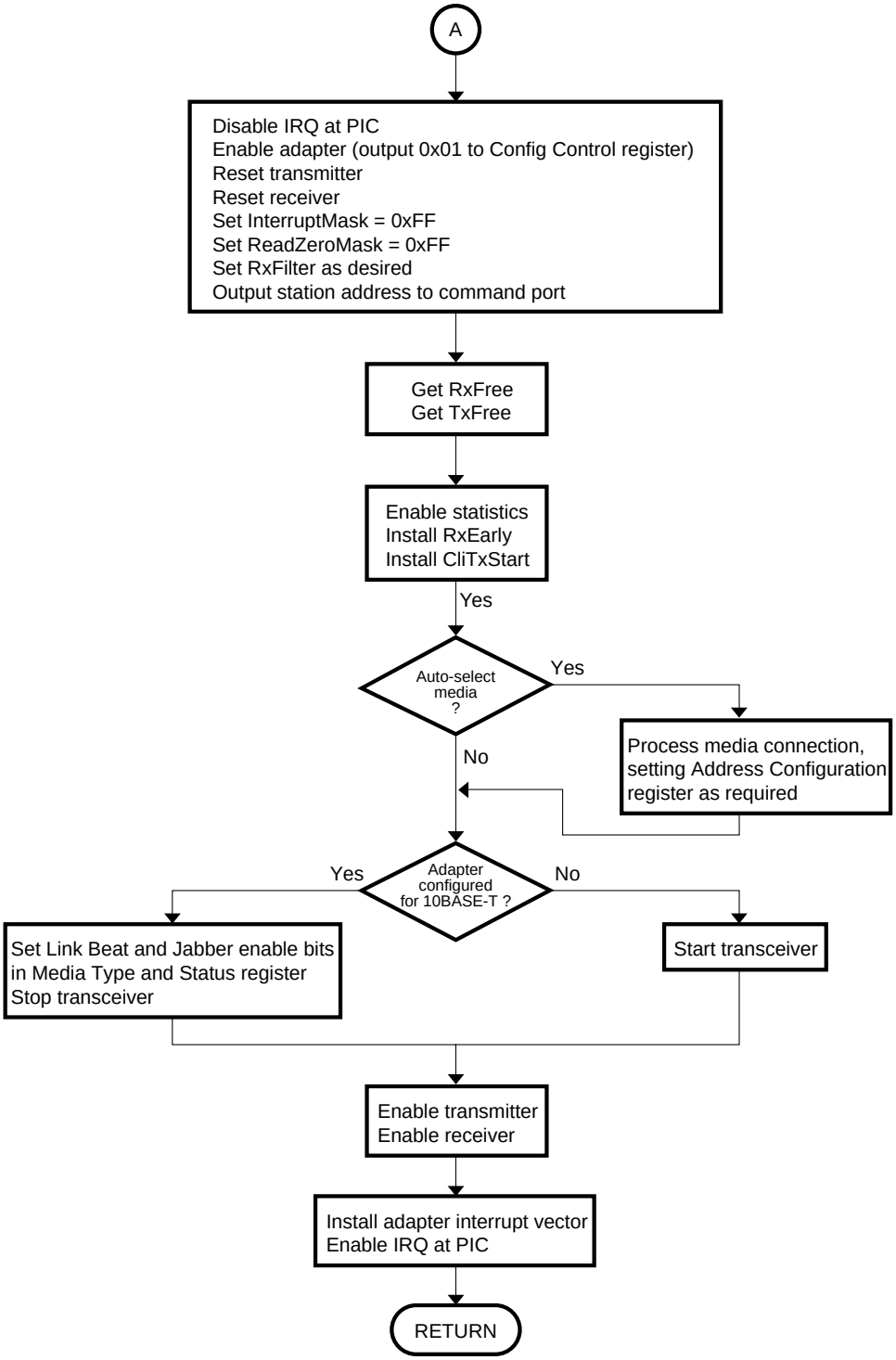


Figure A-2 Initialize Adapter (PCMCIA) (continued)

Get Bus Type

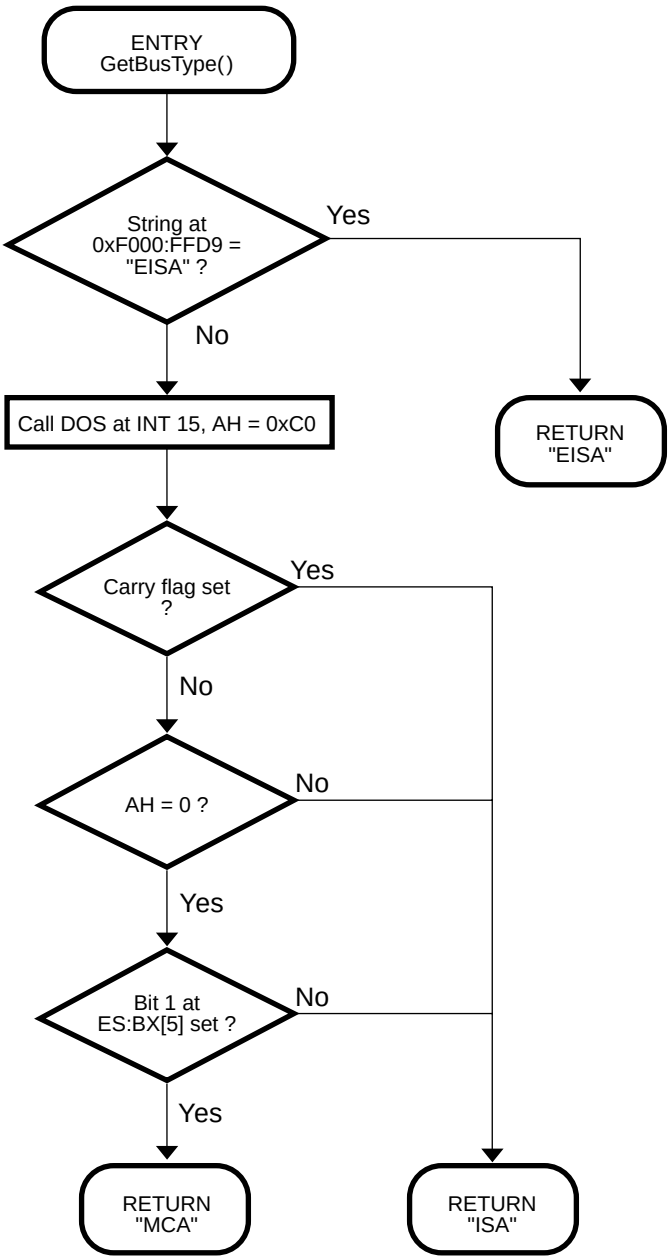


Figure A-3 Get Bus Type

Transmit Frame

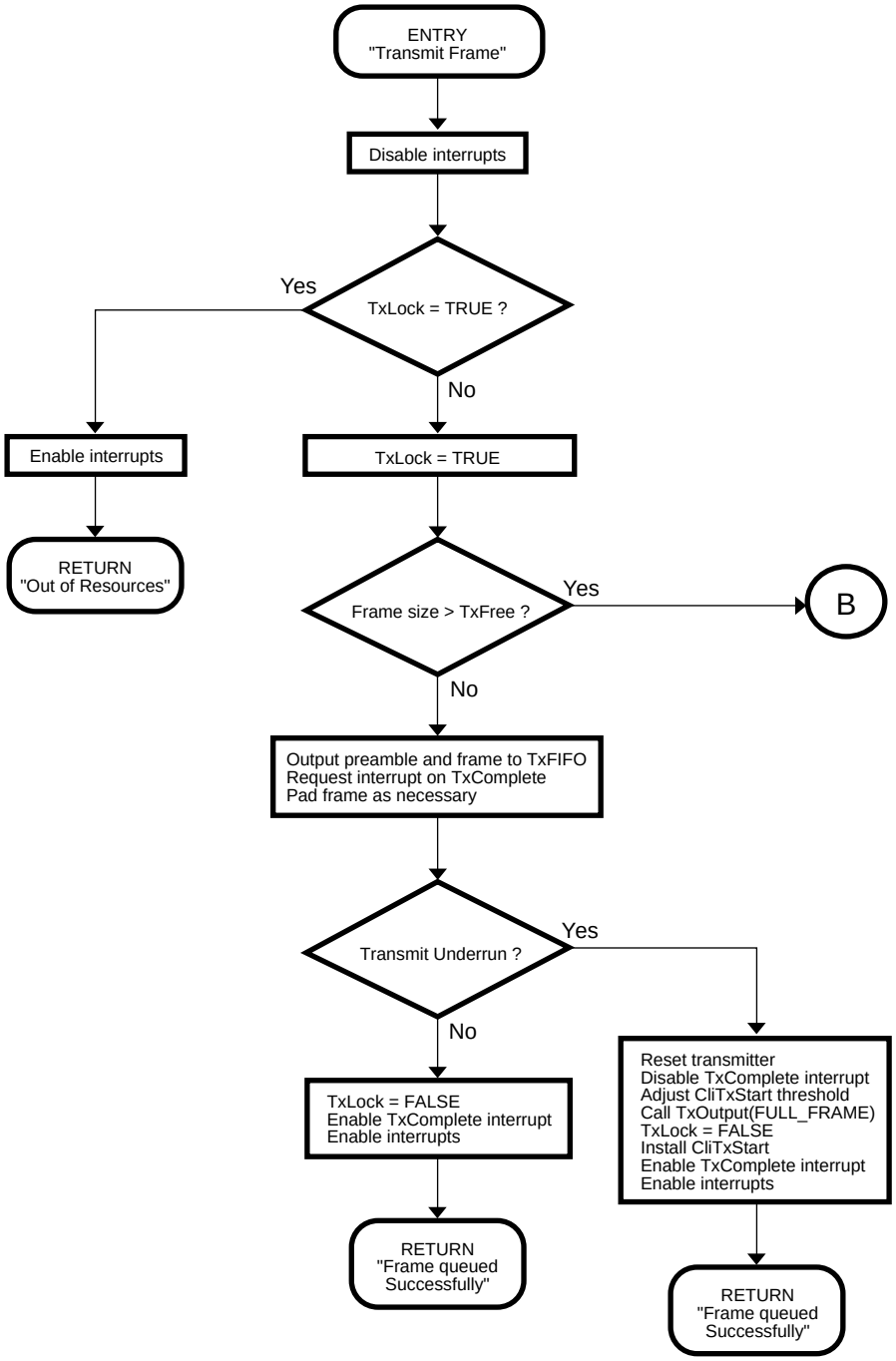


Figure A-4 Transmit Frame

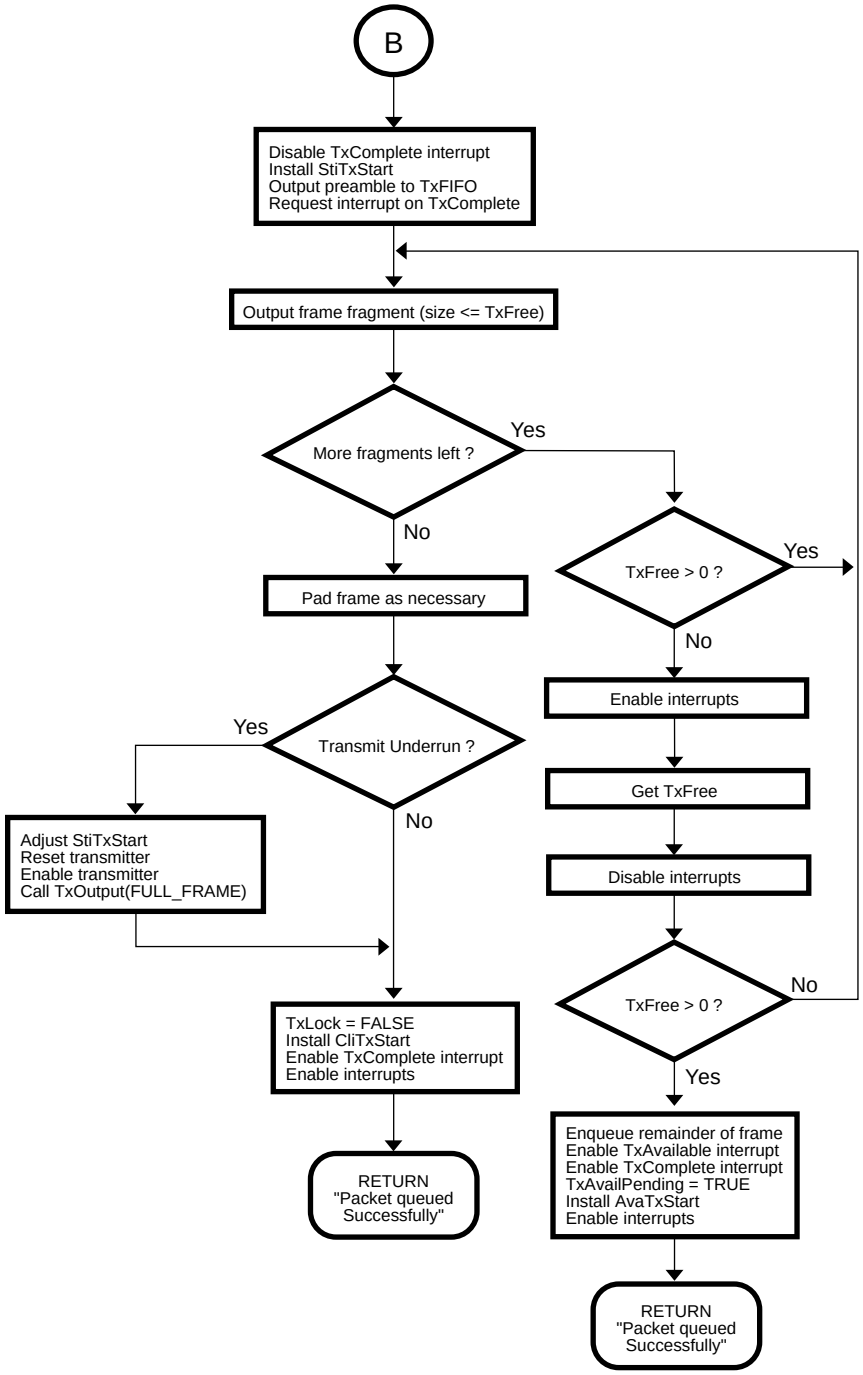
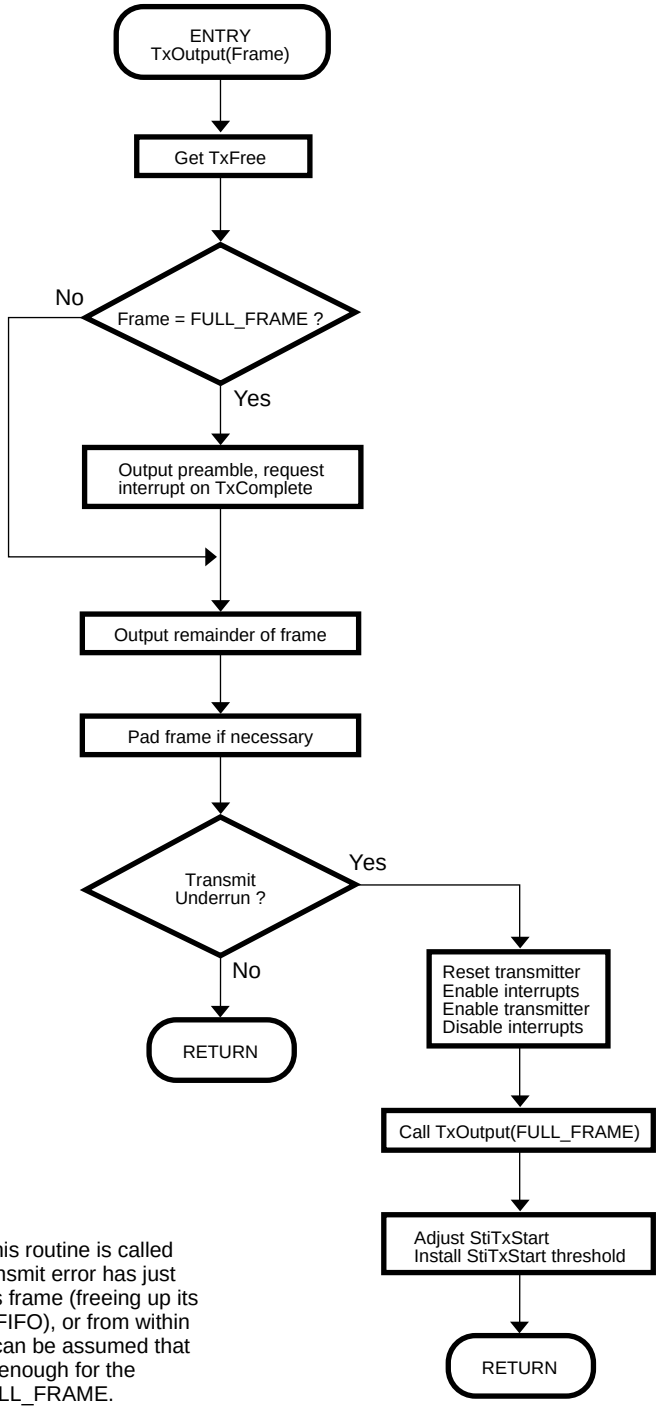


Figure A-5 Transmit Frame (continued)

Transmit Output



NOTE: Since this routine is called only when a transmit error has just occurred for this frame (freeing up its space in the TxFIFO), or from within TXAvailable, it can be assumed that TxFree is large enough for the PARTIAL or FULL_FRAME.

Figure A-6 Transmit Output

Interrupt Service Routines

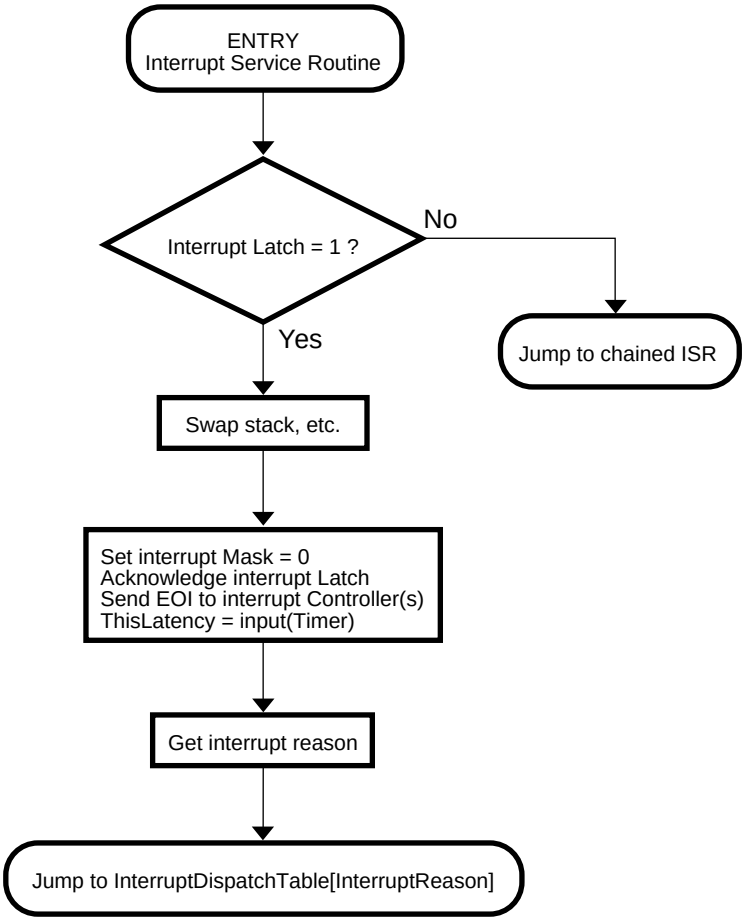


Figure A-7 Interrupt Service Routines (1 of 7)

Figures A-13 through A-18 are the routines pointed to by the entries in Figure A-12.

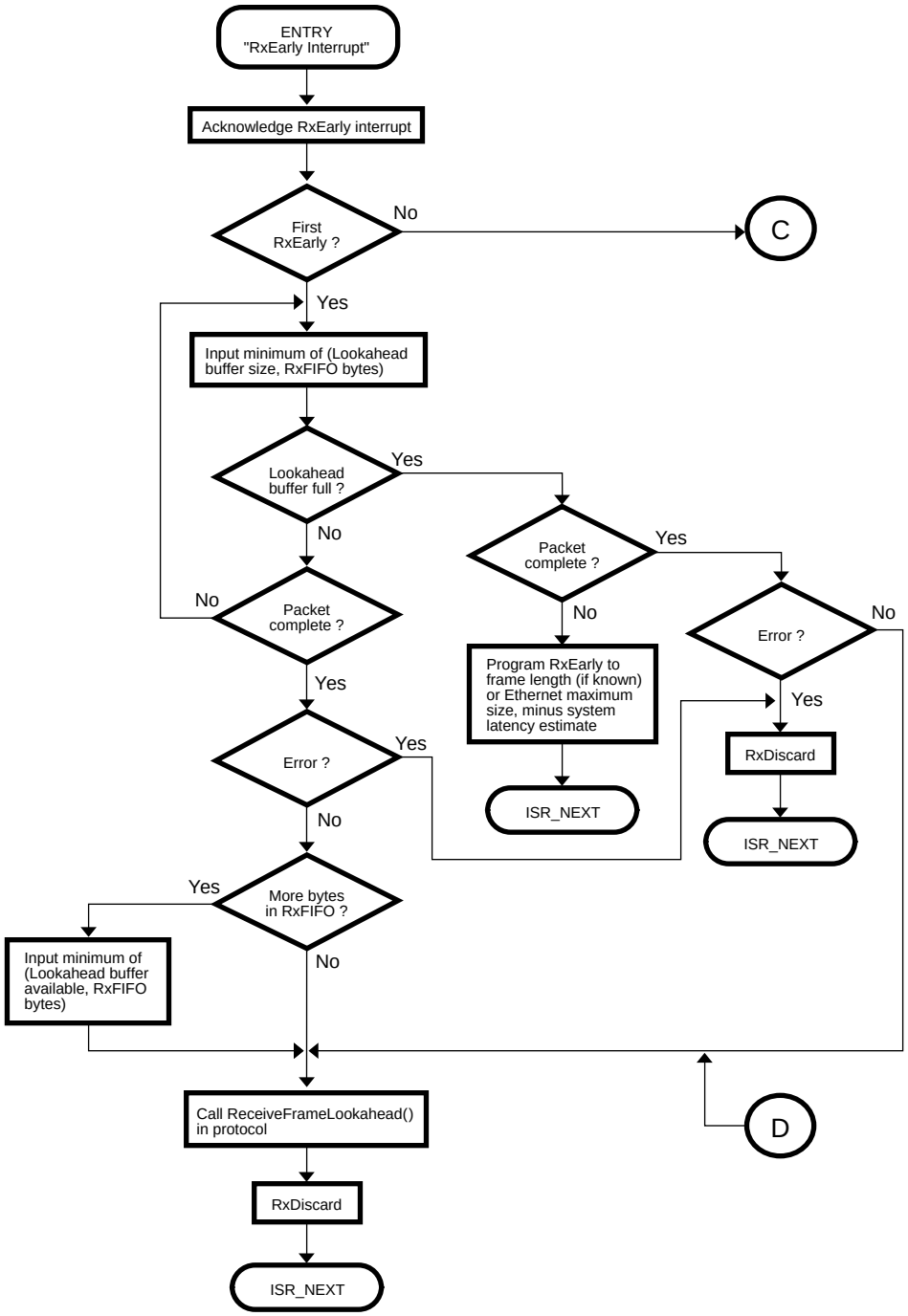


Figure A-8 Interrupt Service Routines (2 of 7)

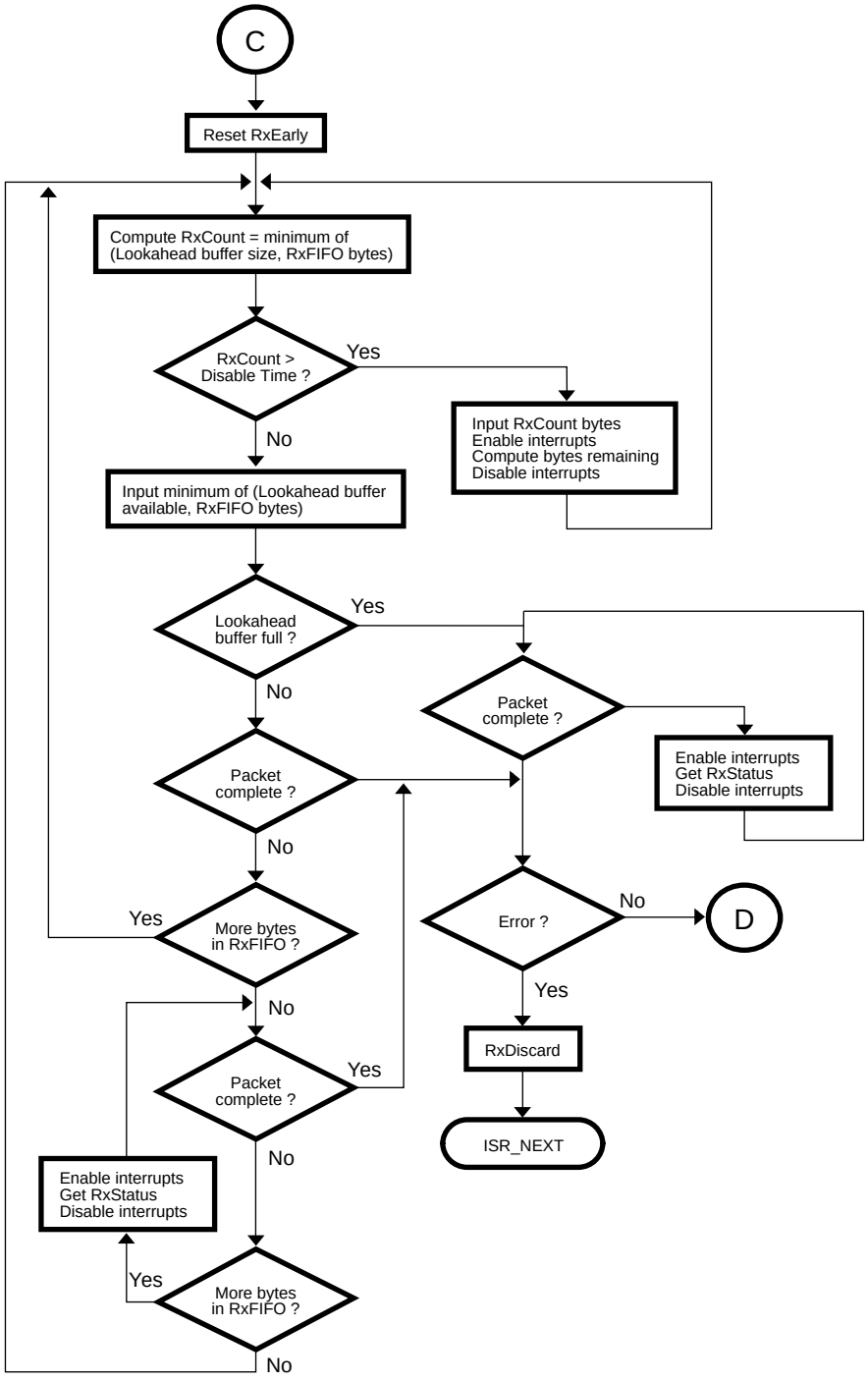


Figure A-9 Interrupt Service Routines (3 of 7)

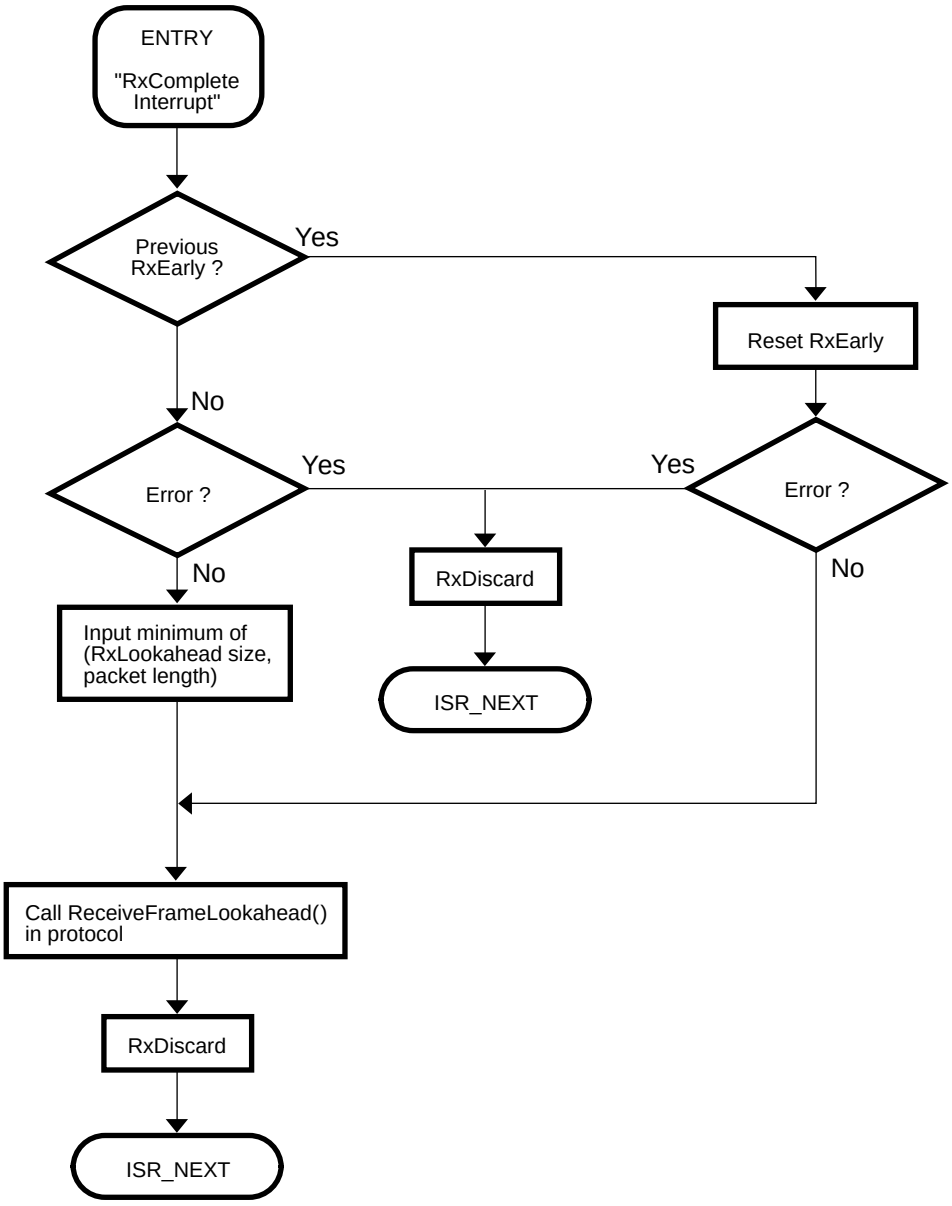


Figure A-10 Interrupt Service Routines (4 of 7)

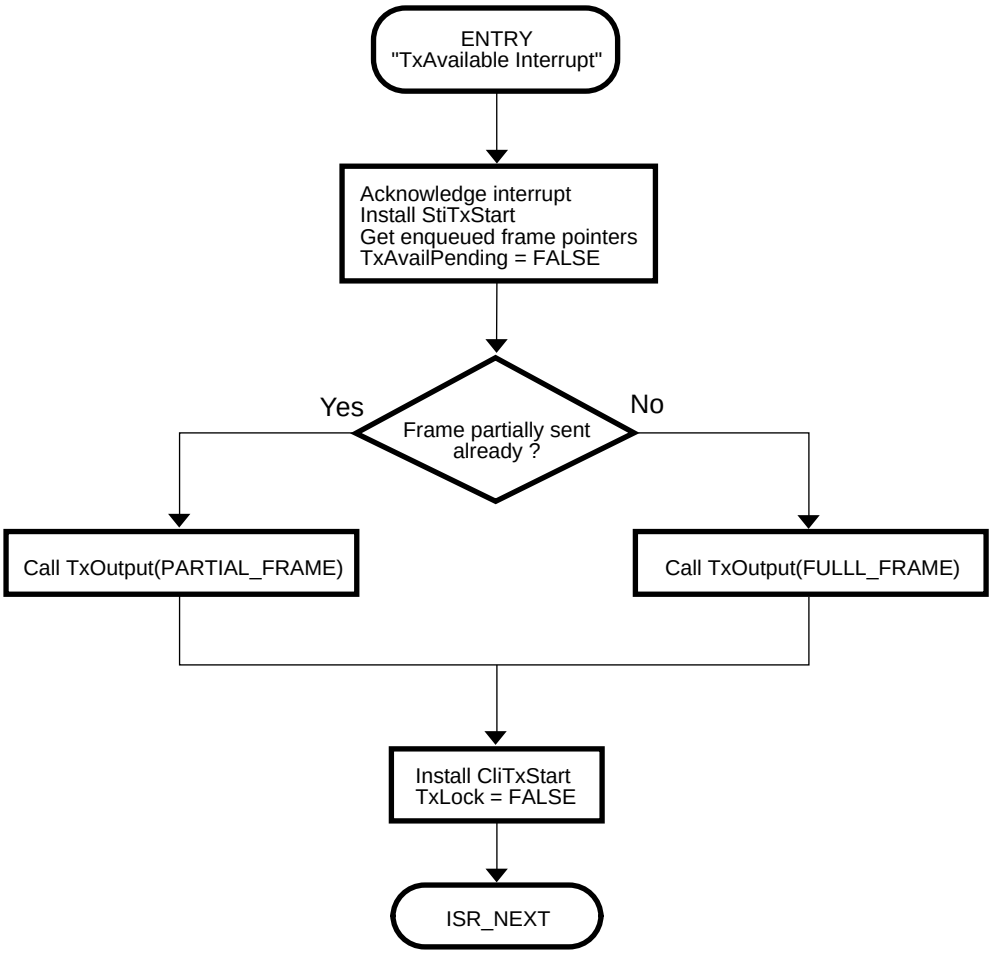


Figure A-11 Interrupt Service Routines (5 of 7)

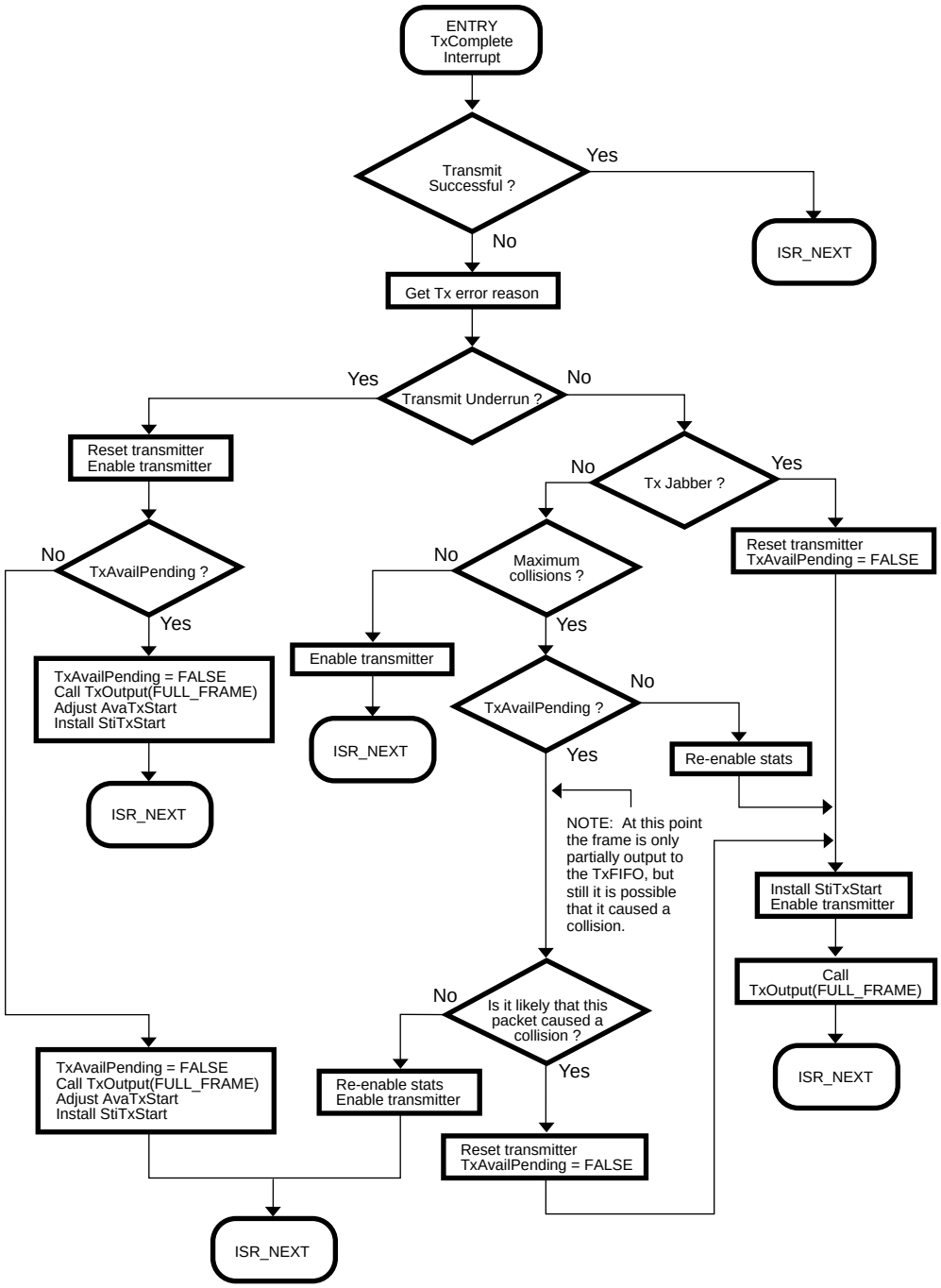


Figure A-12 Interrupt Service Routines (6 of 7)

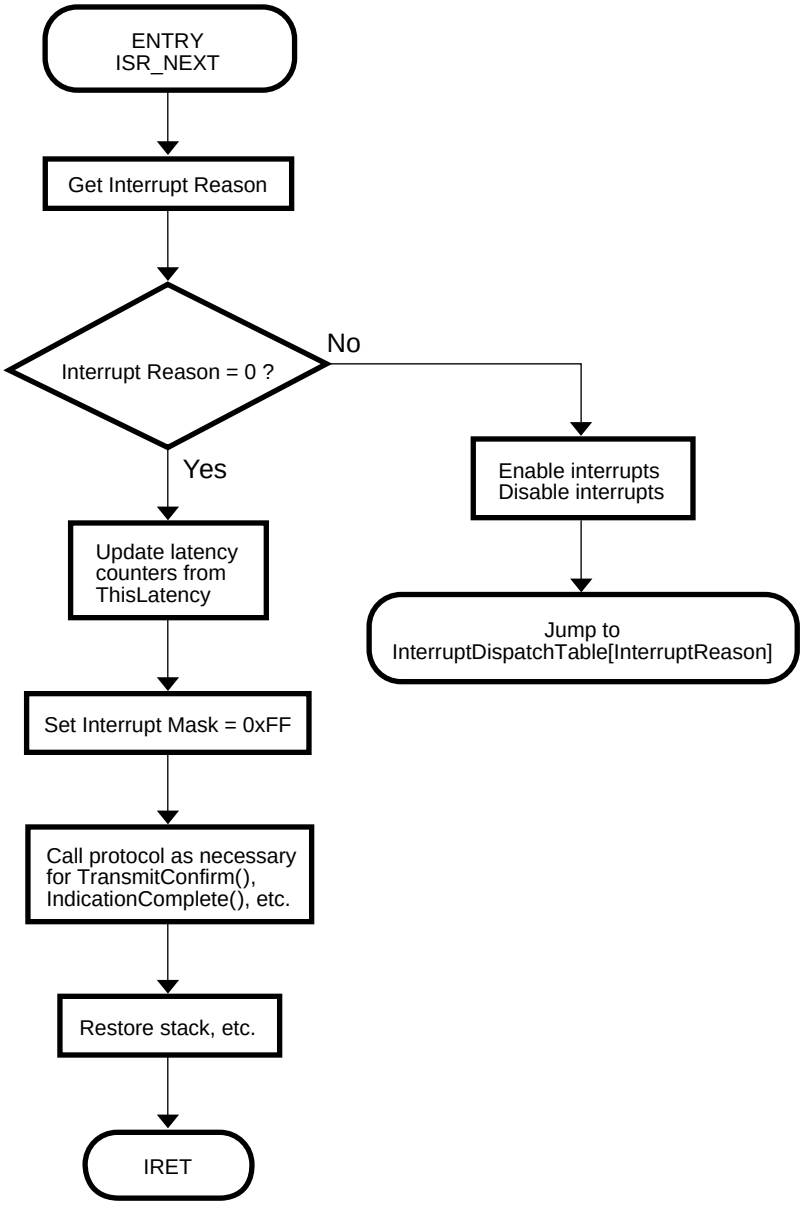


Figure A-13 Interrupt Service Routines (7 of 7)

Timer Interrupt

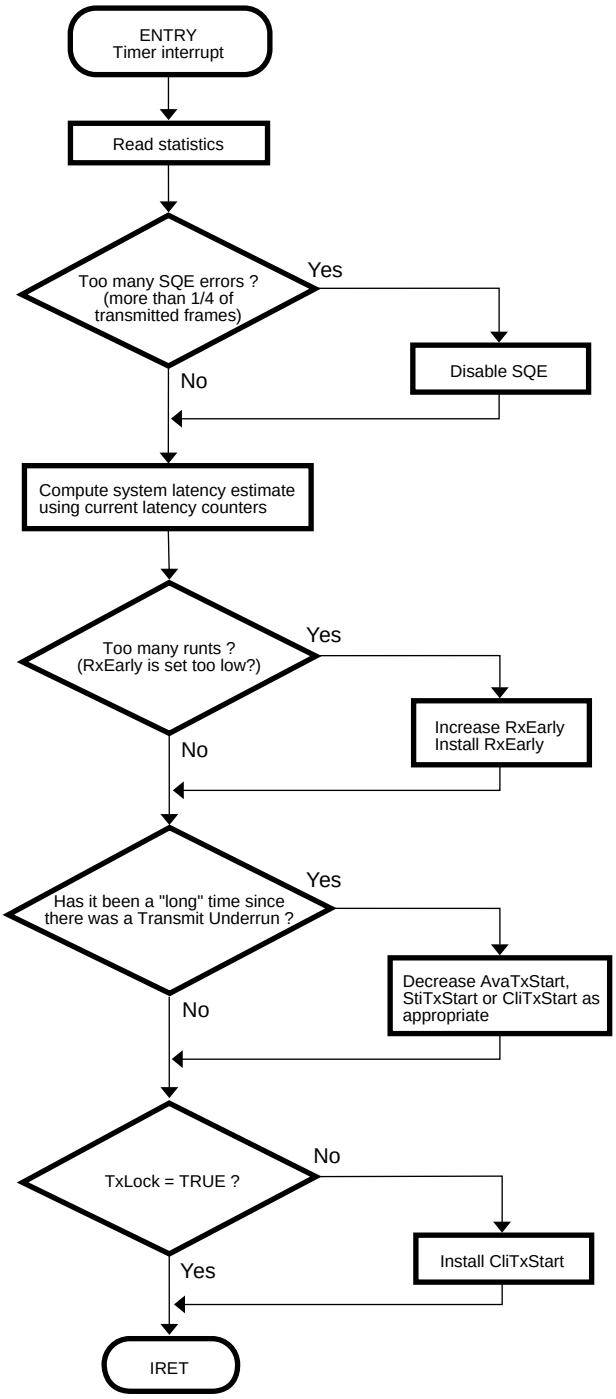


Figure A-14 Timer Interrupt