EtherLink® III Parallel Tasking™ ISA, EISA, Micro Channel®, and PCMCIA Adapter Drivers Technical Reference

Members of the 3Com EtherLink III family of adapters

For 3Com User Group Information
1-800-NET-3Com
or your local 3Com office

Manual Part No. 09-0398-002B
Published August 1994. Printed in the U.S.A.

3Com Corporation reserves the right to revise this publication 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 guide 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 manual at any time.

If you are a government agency, then this software and documentation is 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) of the 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's standard commercial agreement for the Software. Unpublished rights reserved under the copyright laws of the United States.

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

IBM, OS/2, Micro Channel, and AT are registered trademarks of International Business Machines Corporation. Other brand and product names may be the registered trademarks or trademarks of their respective holders.

CONTENTS

Adapter Configuration and Enable

ISA/EISA/MCA/PCMCIA Bus Interfaces

External Configuration Options

Adapter Differences

Driver Routine Flowcharts

FIGURES

TABLES

Chapter 1
Introduction

Overview

This technical reference describes the basic architecture of the 3Com® EtherLink® III Parallel Tasking™ 16- and 32-bit adapters. These adapters are listed in Table 1-1 along with their bus types, cable specifications, and connector types.

Table 1-1. 3Com EtherLink III Parallel Tasking Adapters

Bus Type	Adapter Number	Connectors per Adapter	Cable Specification	Connector Type
ISA	3C509-TPO 3C509B-TPO	One connector	10BASE-T	RJ-45
	3C509-TP 3C509B-TP	Two connectors	10BASE-T Thick coax	RJ-45 AUI
	3C509-Coax 3C509B-Coax	Two connectors	Thick coax Thin coax	AUI BNC
	3C509-COMBO 3C509B-COMBO	Three connectors	10BASE-T Thick coax Thin coax	RJ-45 AUI BNC
EISA	3C579	Two connectors	Thick coax Thin coax	AUI BNC
	3C579-TP	Two connectors	Thick coax 10BASE-T	AUI RJ-45
MCA	3C529	Two connectors	Thick coax Thin coax	AUI BNC
	3C529-TP	Two connectors	Thick coax 10BASE-T	AUI RJ-45
PCMCIA	3C589-TP 3C589B-TP	One connector	10BASE-T	RJ-45
	3C589-COMBO 3C589B-COMO	Two connectors	10BASE-T Thin coax	RJ-45 BNC

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

These adapters are part of the EtherLink III family of high-performance 16- and 32-bit adapters. These adapters interface with the ISA bus, EISA bus, Micro Channel® architecture (MCA) bus, and PCMCIA bus. This technical reference describes all four bus versions. The "B" in the ISA and PCMCIA adapter numbers indicates that these adapters are part of the second generation of the Parallel Tasking™ EtherLink III technology.

NOTE: *Unless otherwise stated, "EtherLink III adapter" refers to all the adapters listed in Table 1-1.*

Information is the same for all adapters in the family except when a section heading specifically limits the information to a group of adapters or to one adapter (for example, by headings such as **ISA/EISA [3C509, 3C579]** or **PCMCIA [3C589]).** When differences occur in the middle of a section, the heading **All** indicates that the subsequent information is common to all the adapters.

Features of the adapter include:

- 3Com-designed Ethernet controller, encoder/decoder, 10BASE-T transceiver, and host interface integrated into one ASIC
- 4 K packet buffer (3C509, 3C529, 3C579, 3C589)
- High performance in client/server applications
- 16-bit ISA bus path
- 16- and 32-bit EISA bus path
- 16- and 32-bit MCA bus path
- 16-bit PCMCIA bus path
- 3C509B and 3C589B only
  - 8 K packet buffer (future versions may support more than 8 K packet buffer)
  - VCO (voltage-controlled oscillator)
  - ISA Plug and Play (compatible for auto-detection and configuration) (3C509B only)
  - Power reduction
  - 8-bit and 16-bit PCMCIA bus path (3C589B only)
- Boot ROM socket, 32 K x 8 (3C509, 3C509B, 3C579, 3C529)

NOTE: *The adapter is optimized to perform best in a client/server with processors such as the 80286-16 MHz and above.*

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.

Chapter 2
Architectural Overview

EtherLink III adapters are designed to be efficient, low-latency network adapters optimized for server and client environments. This has resulted in an adapter with 4 K of buffer space on the board (for 3C509, 3C529, 3C579, and 3C589), which appears as two dedicated FIFOs to the host (one for transmit and one for receive). For 3C509B and 3C589B, there is at least 8 K of buffer space on the board with the two dedicated FIFOs 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 adapter while using only 16 bytes of I/O space.

Refer to Figures 2-1 through 2-4 for block diagrams of the EtherLink III adapters.

ISA/EISA (3C509, 3C509B, 3C579)

The EEPROM sets all configuration options on the board, eliminating the need for jumpers. Methods for EISA auto configuration are supported. Board type and revision number are electronically readable. Interrupt level, I/O base address, and the decode address of the optional remote boot PROM are among the configuration options. A key provides security against accidental reconfiguration. Plug and Play automatic configuration is supported in the 3C509B.

MCA (3C529)

The EEPROM sets the Adapter ID on the board. Board type and revision number are electronically readable. Interrupt level, I/O base address, and the decode address of the optional remote boot PROM are among the configuration options that are set in the programmable option select (POS) registers.

PCMCIA (3C589, 3C589B)

The PCMCIA Card Information Structure (CIS) provides information about how the 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.

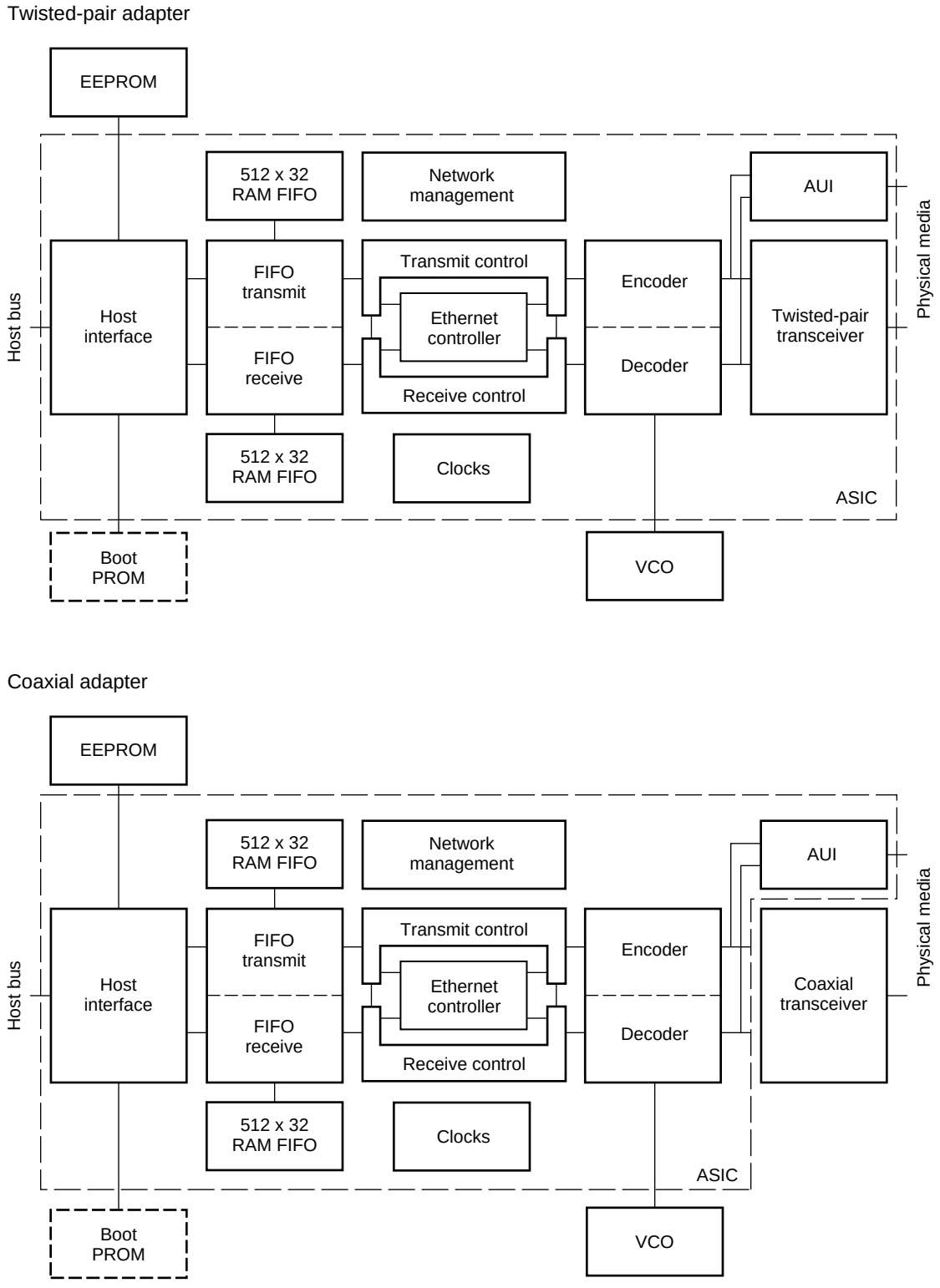


Figure 2-1. Block Diagrams for ISA, MCA, EISA (3C509, 3C529, 3C579)

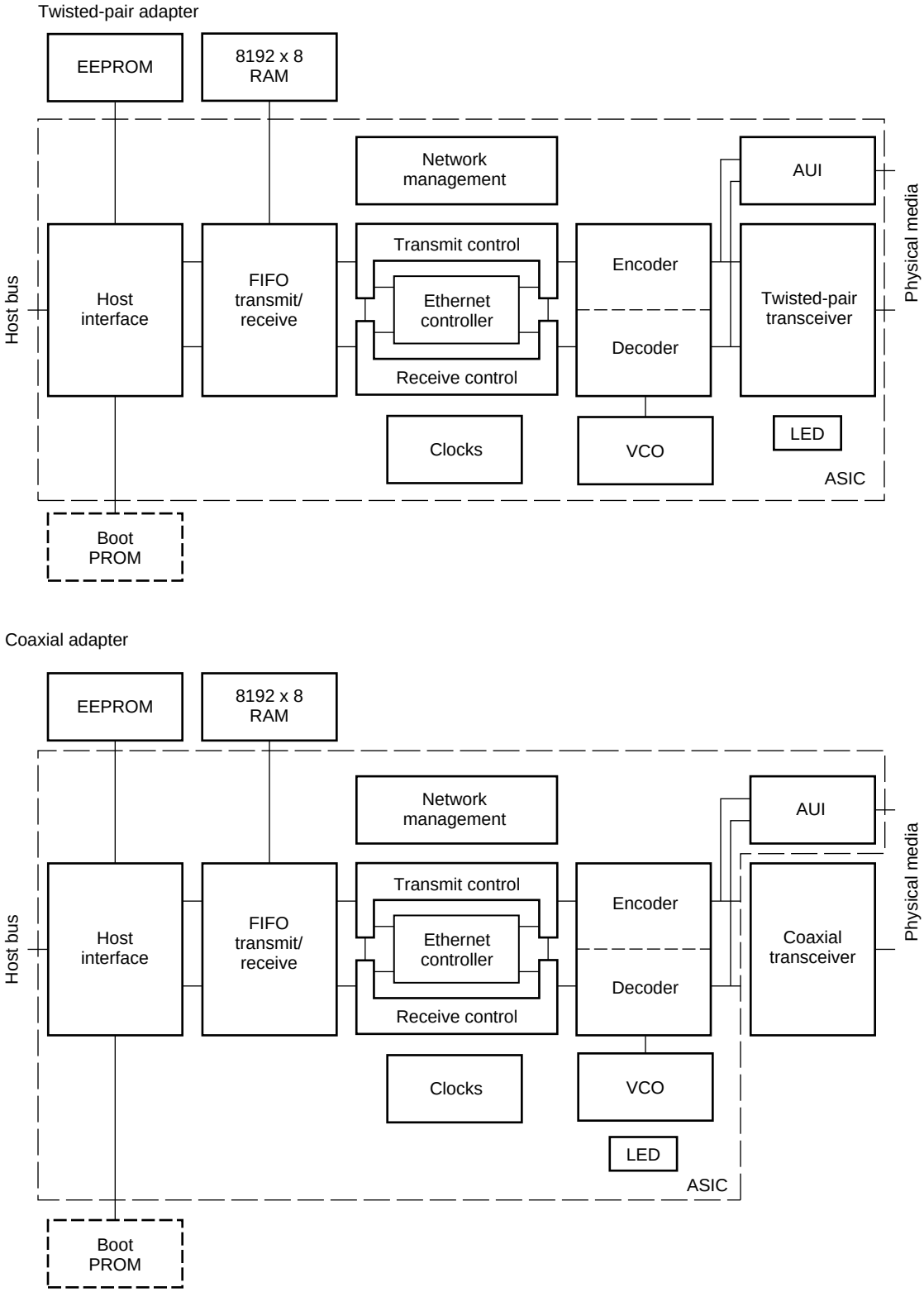


Figure 2-2. Block Diagrams for ISA (3C509B)

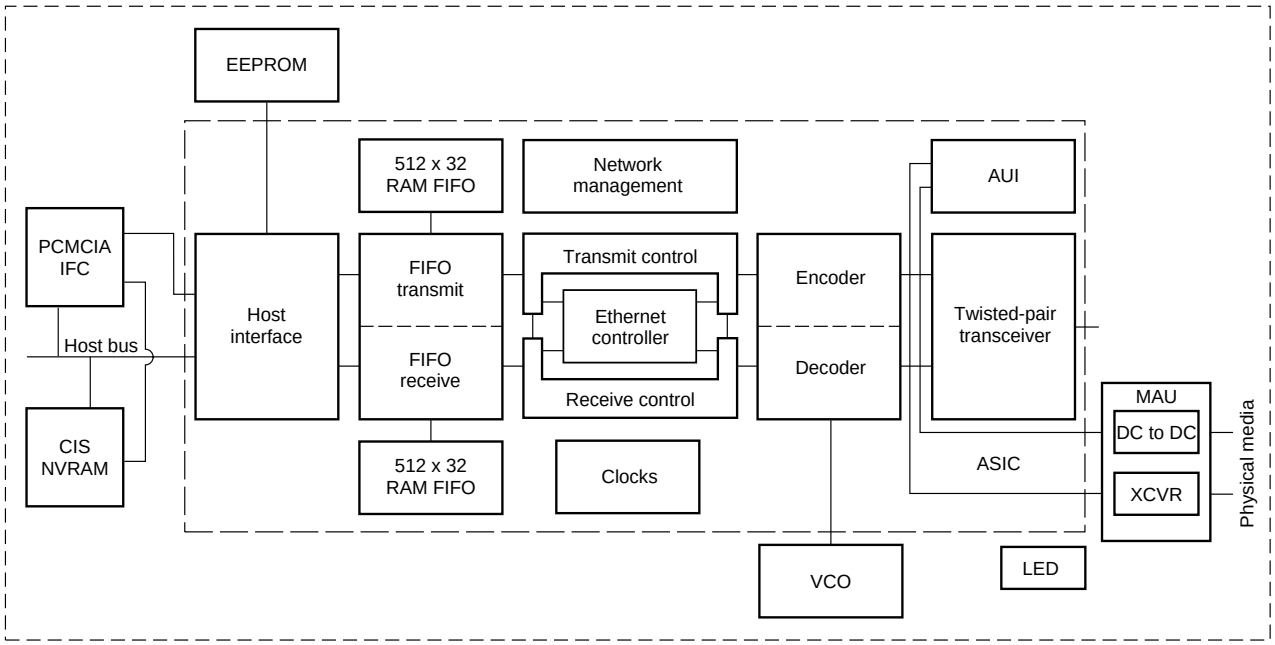


Figure 2-3. Block Diagram for PCMCIA (3C589)

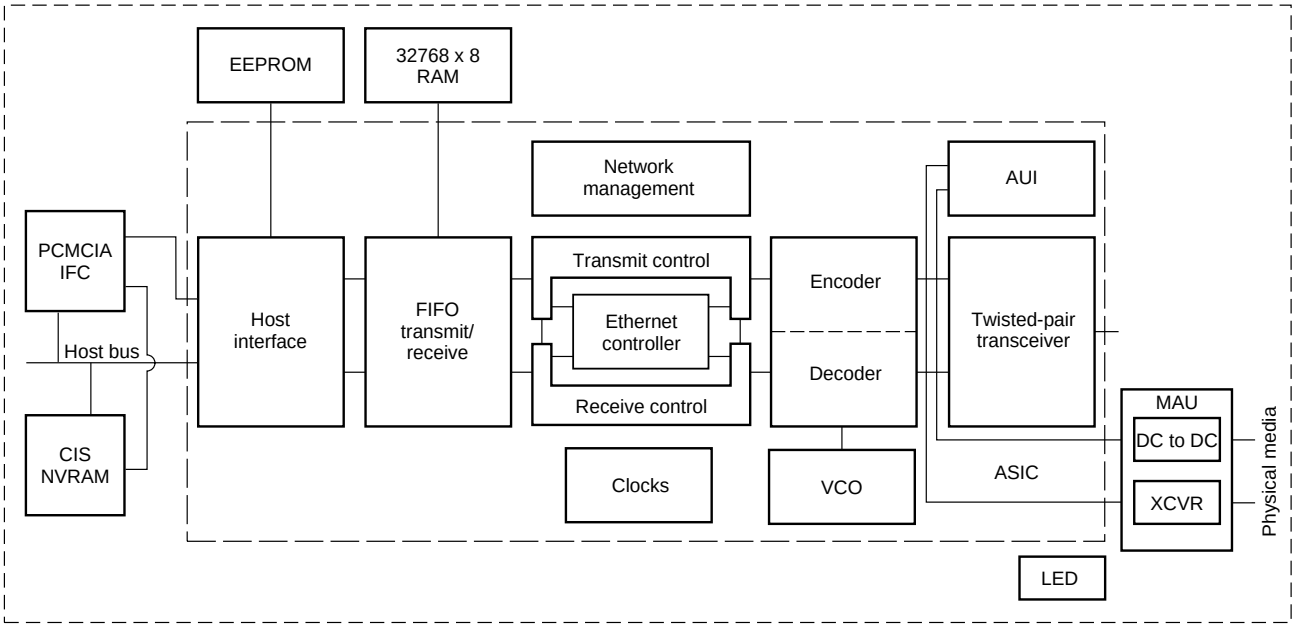


Figure 2-4. Block Diagram for PCMCIA (3C589B)

Chapter 3
Data Structures

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

The adapter'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 adapter, 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 adapter.

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 ISA, EISA, MCA, and PCMCIA (3C509, 3C509B, 3C579, 3C529, 3C589, 3C589B) 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.

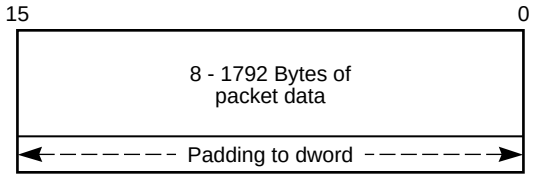


Figure 3-1. Receive Packet Structure

- 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 ISA, EISA, MCA, and PCMCIA (3C509, 3C509B, 3C579, 3C529, 3C589, 3C589B) as it is stored in the FIFO.

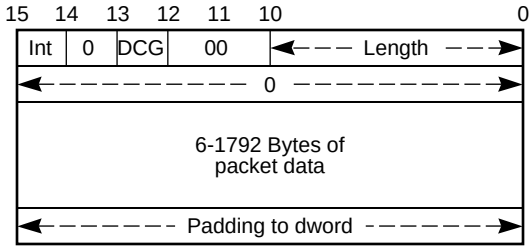


Figure 3-2. Transmit Packet Structure

- 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 (bit 13 for 3C509B). 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.

Chapter 4
Description of Operation

The data transfer mode for the EtherLink III adapter 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 adapter a byte, word, or double word at a time through the PIO Data Read register. Similarly, transmit data can be written to the adapter 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 3C509B and 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 adapter 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 adapter 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 adapter 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 adapter 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 adapter.

Disable CRC Generation (3C509B, 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 adapter 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 adapter. 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.

NOTE: *Underruns occur only when the packet is being copied to the adapter. 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).*

Chapter 5
Window Set

The EtherLink III adapter 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.

PCMCIA (3C589, 3C589B)

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.

Port offset	Write function: High byte	Write function: Low byte	Read function: High byte	Read function: 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

MCA (3C529)

Several of the bits in Window 0 are write accessible from the POS registers. These bits are described in more detail in Chapter 7, “Adapter Configuration and Enable.”

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.

Port offset	Write function		Read function	
	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.

Port offset	Write function		Read function	
	High byte	Low byte	High byte	Low byte
0E	Command		Window (0-7)	Status
0C				
0A				
08				
06				
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

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	
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 control*	
Internal configuration	

* 3C509B only

Figure 5-4. FIFO Management

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	
Net diagnostic	
FIFO diagnostic	

Figure 5-5. Diagnostics

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 by ISA and PCMCIA (3C509B, 3C589B).

Write function (All)

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

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

High byte	Low byte
Window (0-7)	Status
Read zero mask	
Interrupt mask	
RX filter (lower 4 bits only)	
RX early threshold	
TX available threshold	
TX start threshold + 4	

Read function (3C509B, 3C529, 3C589B)

Port offset	High byte	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 start threshold	

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 by ISA and PCMCIA (3C509B, 3C589B). Refer to the section “Statistics Registers” in Chapter 6 for more information.

Port offset	Write function		Read function	
	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	Receive overruns	Late collisions on transmit	Receive overruns	Late collisions 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

Chapter 6
Register Definitions

This chapter describes the function and use of each register needed in normal operation of the EtherLink III adapter. 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.

ISA/PCMCIA (3C509B, 3C589B)

In general, both 8- and 16-bit accesses are allowed to all registers. Certain 8-bit registers must be read/written as 8 bits. These will be documented where appropriate. When a 16-bit register is accessed using 8-bit operations, the low byte access must be followed by the high byte access, with no other accesses to the adapter in between. Certain exceptions are allowed, which will be noted where appropriate. Any side-effects are documented in the register description.

Command Register

Function: Issues commands to adapter.
Location: All windows/Port 0E
Type: Write only
Size: 16 bits

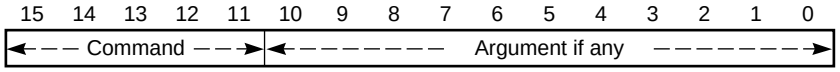


Figure 6-1. Command Register

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.

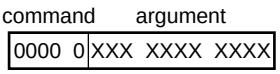
Command Register Conventions

The Command register controls many adapter functions. For example, a command is used to switch windows. In general, commands must be written as a word, or for 3C509B only, low-byte/high-byte (executed when the high byte is written). 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*

0000 0	000 0000 0000

(3C509, 3C579, 3C589)

0000 0	000 00XX XXXX

(3C509B, 3C529, 3C589B)

ISA/EISA/PCMCIA (3C509, 3C579, 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 adapter again. The argument to this command must be all zeros.

ISA/MCA/PCMCIA (3C509B, 3C529, 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. This command, on the 3C529, requires 16 clocks to execute, and the Command-in-Progress bit in the Status register is set while this command is in progress.

NOTE: *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 ISA/EISA/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 adapter is in test mode. ISA (3C509B) adapters will return to the AUTOINIT state. Plug and Play configuration is also placed in a reset state.
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

(3C509B, 3C529, 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.

ISA/MCA/PCMCIA* (3C509B, 3C529, 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*

0100 0	000 0000 0000

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 adapter 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

0100 1	000 0000 0000

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

0101 0	000 0000 0000

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	(3C509B, 3C529, 3C589B)
0101 1	X00 0000 XXXX	(3C509B, 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 adapters.

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

ISA/MCA/PCMCIA (3C509B, 3C529, 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).

3C509B and 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.

ISA/PCMCIA (3C509B, 3C589B)

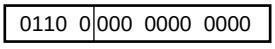
TX Reset is required after a Power Up command is issued while the chip is in a Power Down Full state.

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.

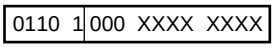
1 = Transmit will be aborted immediately. A bad CRC is not guaranteed to be generated. See discussion above.

Request Interrupt

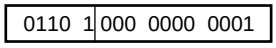


This command sets the Interrupt Requested bit in the Status register, causing an interrupt whenever that interrupt bit is unmasked.

Acknowledge Interrupt



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 adapter 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** 

  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 1000 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 1000 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 1000 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 1000 0001 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 1000 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 1000 0100 0000

  This command turns off the Status register bit.

- **Update Statistics** 0110 1000 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 0000 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 adapter, 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 1000 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.

NOTE: *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 adapter 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.

ISA/EISA/PCMCIA (3C509, 3C579, 3C589)

These adapters 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.

ISA/MCA/PCMCIA (3C509B, 3C529, 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 adapter 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 adapter 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 adapter. 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 adapter. At power-up, statistics collection is disabled. The adapter 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.

Set TX Reclaim Threshold

1100 0	XXX ZZZZ ZZZZ

This command (for MCA [3C529] only) specifies the number of bytes that must be transmitted into the network before the FIFO memory space occupied by the transmitting packet is reclaimed as Free Transmit bytes. By default, the 3-bit argument field is 111, which disables this function. When disabled, the entire packet is transmitted before the FIFO memory space occupied by the transmitting packet is reclaimed as Free Transmit bytes. The command argument is set in increments of 256 bytes. The minimum allowed value is 001. A value of 000 may cause intermittent behavior. Once the reclaim threshold is crossed, the Free Transmit bytes continue to increase as each byte is actually transmitted onto the network. Also, once the reclaim threshold is crossed, a collision will cause the packet to be aborted with a transmit error (noted in TX Status bit 1), the complete packet space is reclaimed as Free Transmit bytes, and a TX Complete interrupt is generated.

Power-Management Commands (3C509B, 3C589B)

The following three power-management commands are new with 3C509B and 3C589B adapters, and are ignored by 3C509, 3C579, 3C529, and 3C589 adapters. 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 adapter.

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 adapter 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 adapter is currently powered-up or powered-down. The next access to the adapter 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 adapter (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 adapter 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)

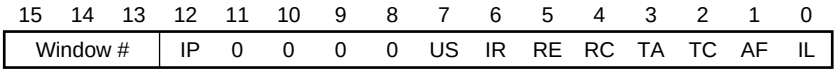


Figure 6-2. Status Register

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.

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

ISA/MCA/PCMCIA (3C509B, 3C529, 3C589B)

NOTE: *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 adapter 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.

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

Bit 1 Adapter Failure†

An error occurred that the adapter 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.

ISA/MCA (3C509B, 3C529)

The host must issue the appropriate Reset command to clear this condition and recover. The host should mask the reset to Host Reset, AISM Reset, TPAUI Reset, and EnDec Reset for proper recovery, because this logic cannot be the source of an adapter failure.

Bit 2 TX Complete†

The adapter 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.

NOTE: *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 adapter 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 adapter. 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

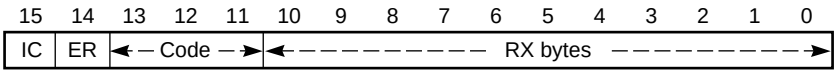


Figure 6-3. FIFO Register

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).

ISA/EISA/PCMCIA (3C509, 3C579, 3C589)

These adapters 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

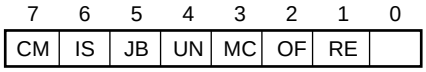


Figure 6-4. TX Status Register

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 (ISA/EISA/PCMCIA [3C509, 3C579, 3C589]: TX Reset required) (ISA/MCA/PCMCIA [3C509B, 3C529, 3C589B]: Network TX Reset and FIFO TX Reset required)
Bit 3	Maximum Collisions
Bit 2	TX Status Overflow
Bit 1	Undefined **(**ISA/EISA/PCMCIA [3C509, 3C509B, 3C579, 3C589, 3C589B])
Bit 1	TX Reclaim (MCA [3C529])
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 adapter 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 adapter, the host can continue to copy the packet to the adapter, since the transmitter is disabled.

NOTE: *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; 32 bits for the EISA and MCA adapters

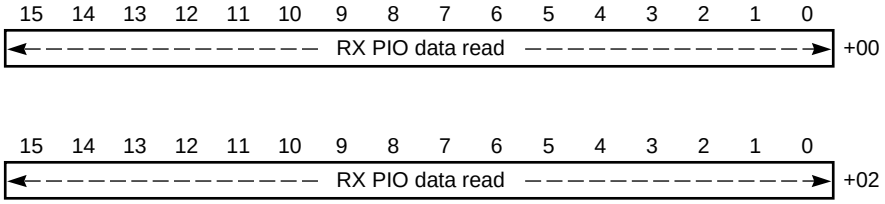


Figure 6-5. RX PIO Data Read Register

Bit Description: Bits 15–0 RX PIO Data Read

ISA/EISA/PCMCIA (3C509, 3C509B, 3C579, 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.

MCA (3C529)

This register is used to read data from the RX FIFO. A dword read pops a dword off the head of 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 are allowed at offset +00 or at offset +02. The 32-bit EtherLink III Micro Channel adapter works in a 16-bit slot, and the software needs no special detection of this circumstance, because the hardware of the bus and adapter takes care of it. If for some reason the driver wanted a 32-bit adapter to behave only as a 16-bit adapter, all data accesses should be offset to +02.

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 byte; 32 bits for the EISA and MCA adapters

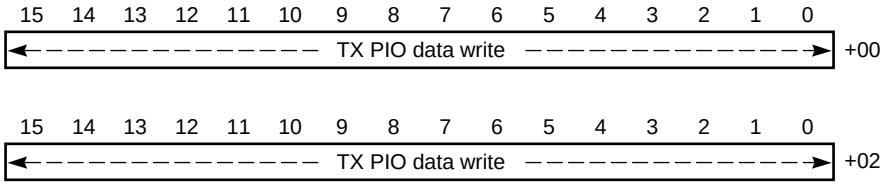


Figure 6-6. TX PIO Data Write Register

Bit Description: Bits 15–0 TX PIO Data Write

ISA/EISA/PCMCIA (3C509, 3C509B, 3C579, 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.

MCA (3C529)

This register is used to write data to the TX FIFO. A dword write pushes a dword onto the tail of the TX FIFO. A word write to either offset +00 or +02 pushes a word onto 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. If the driver wants a 32-bit adapter to behave only as a 16-bit adapter, all data accesses should be offset to +02.

All

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

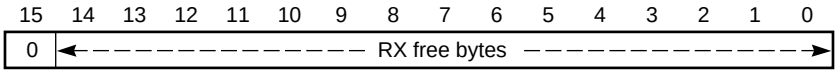


Figure 6-7. Free Receive Bytes Register

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

The actual range depends on the adapter 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

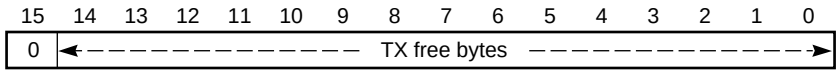


Figure 6-8. Free Transmit Bytes Register

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

The actual range depends on the adapter 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.

NOTE: *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

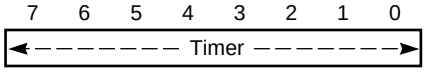


Figure 6-9. Timer Register

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.

ROM Control Register (3C509B only)

Function: Control of the 16 K ROM page being accessed.
Location: Window 3/Port 05
Type: Read/write
Size: 8 bits

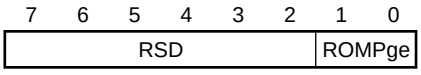


Figure 6-10. ROM Control Register

Bit Description: Bits 7–2 Reserved, read as zero

Bits 1–0 ROM Page

The ROM Control register controls which 16 K ROM page is visible to the host through the ROM space in upper memory. It is ignored except for ROM sizes of 32 or 64 K. Reading these bits returns the current ROM page. Writing them sets the ROM page.

Statistics Registers

The Statistics registers in Window 6 make available various counters maintained by the adapter. 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 adapter 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 adapter 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)

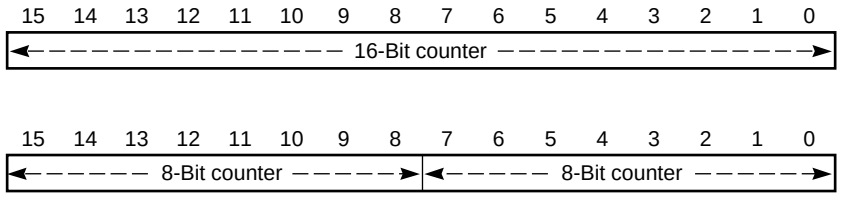


Figure 6-11. Statistics Register

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.**

NOTE: *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
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.

NOTE: *With ISA/EISA/PCMCIA (3C509/3C579/3C589) adapters, 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. 3C529, 3C509B, and 3C589B adapters do not require the Read Zero mask for the interrupt. All types of 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

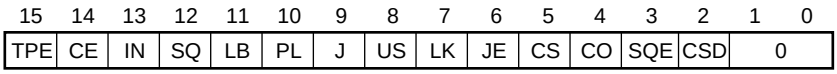


Figure 6-12. Media Type and Status Register

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[1] (3C509B and 3C589B). 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 adapter 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.

1. 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.

Net Diagnostic Port

Function: Supports the network diagnostic.
Location: Window 4/Port 06
Type: Read/write (only certain bits are writable)
Size: 16 bits

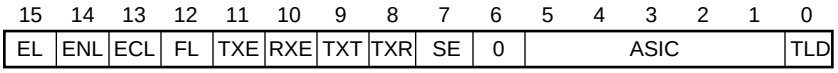


Figure 6-13. Net Diagnostic Port Register

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). 3C509, 3C529, 3C579, 3C589 =1. 3C509B, 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.

NOTE: *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

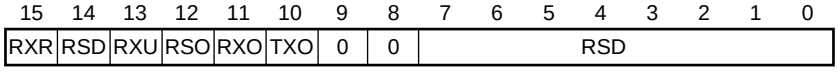


Figure 6-14. FIFO Diagnostic Port Register

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 MCA, ISA, and PCMCIA (3C529, 3C509B, and 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 3C509, 3C529, 3C579, and 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 3C509B and 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

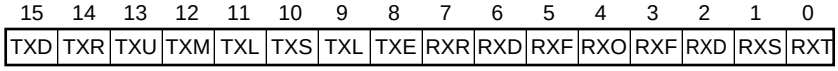


Figure 6-15. Ethernet Controller Status Register

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).

Chapter 7
Adapter Configuration and Enable

EtherLink III adapters 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 3C509, 3C509B, and 3C579 adapters support adapter activation mechanisms for EISA and ISA machines. The 3C589 and 3C589B adapters support the PCMCIA activation mechanisms, and the 3C529 adapter supports the Micro Channel activation mechanisms.

NOTE: *Once activated, the adapter can be deactivated only by a Global Reset command.*

Automatic Configuration at Power-on Reset

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

ISA/EISA (3C509, 3C579)

After reset (power-on reset, Configuration Control register bit 2, Global Reset command, or ID Global Reset command) the adapter copies the eighth, ninth, and third words of the EEPROM into the Address Configuration register, the Resource Configuration register, and the Product ID register. The I/O Base Address configuration is also written at this time.

ISA (3C509B)

After reset (power-on reset, Configuration Control register bit 2, Global Reset command, or ID Global Reset command) the adapter copies the eighth, ninth, and third words of the EEPROM into the Address Configuration register, the Resource Configuration register, and the Product ID register, as well as loading the Internal Configuration register from words 12h and 13h.

MCA (3C529)

After reset (power-on reset, Configuration Control register bit 2, Global Reset command, or ID Global Reset command) the adapter copies the ninth and third words of the EEPROM into the Resource Configuration register and the Adapter ID register.

The 3C529 adapter supports the POS registers of the Micro Channel architecture. Several Window 0 register bits loaded from the on-board configuration EEPROM that are read/write on the 3C509 are read-only in Window 0 for the 3C529 adapter. These register bits are read/write in the POS registers of the 3C529. The following register bits are loaded from the EEPROM on the 3C509 but are POS bits in the 3C529:

- IOBASE
- ROM SIZE
- ROM BASE
- XCVR
- IRQ

PCMCIA (3C589, 3C589B)

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 adapter's I/O interface and then write the appropriate data to the following registers: Address Configuration, Resource Configuration, and, optionally, Product ID.

ISA Activation Mechanism (3C509, 3C509B)

The ISA activation mechanism is enabled when the I/O BASE is set to an ISA base address (200h–3E0h).

ISA (3C509B)

ISA activation is enabled when the I/O Base is set as above and the ISA Activation Select bit (Internal Configuration register) is set to either "Both mechanisms enabled" or "ISA contention only."

ISA (3C509, 3C509B)

In this mode, memory read accesses to the boot PROM are always enabled after automatic configuration is completed, but I/O accesses to the adapter's I/O base address are disabled until the ID Sequence State machine (IDS) activates them. Refer to Figure 7-1.

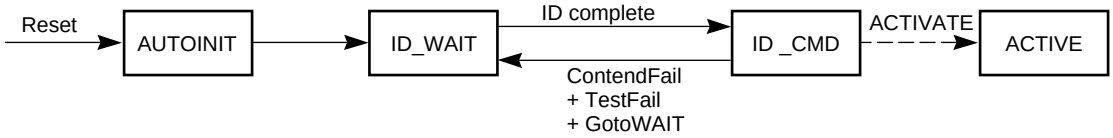


Figure 7-1. ID Sequence State Machine (IDS)

NOTE: *The adapter is not visible to the software until after the automatic configuration logic has finished reading the adapter configuration EEPROM. This takes 310 µs after a global reset. A global reset can be caused by power-on reset (RESETDRV), a Global Reset command written to the Command register (port 0E), an ID Global Reset command written to the ID command port, or setting the RST (reset) bit (bit 2) in the Configuration Control register (port 04).*

After the automatic configuration is completed, the IDS is in its initial state (ID_WAIT) (see Figure 7-1), and it monitors all write access to I/O port 01x0h, where x is any hex digit. If a zero is written to any one of these ports, then that address is remembered and becomes the ID port. A second zero written to that port resets the ID sequence to its initial state. The IDS watches for the ID sequence to be written to the ID port.

The ID sequence is a sequence of 255 bytes defined by the following algorithm:

```
              mov     cx,   0FFh
              mov     dx,   IDport
              mov     al,   0FFh
@@1:          out     dx,   al
              shl     al
              jnc     @@2
              xor     al,   0CFh
@@2:          loop    @@1
```

The sequence starts with the value 0FFh and ends with 98h (if the adapter is in test mode, the sequence ends with 69h, the eighth byte). If at any point in the sequence an incorrect value is written, the sequence is reset to its initial value of 0FFh. Since the value zero never appears in the ID sequence, writing a zero always resets the sequence and the ID port. When the complete sequence has been written, the IDS enters the command state.

In the ID_CMD state the IDS responds to both I/O reads and writes to the ID port. I/O writes to the ID port are interpreted as commands. The following commands are supported:

00 to 7F	Go to ID_WAIT state. Wait for next ID sequence.
80 to BF	Read EEPROM word n. Address n is the last 6 bits of the command. EEPROM data is read into the EEPROM data register.

(Refer to the EEPROM Command register description for read/write timing requirements.)

C0 to CF	Global reset. Resets adapter to the same state as power-on reset (POR).
D0 to D7	Set adapter tag register to *n*, where *n* is last three bits of the command. If adapter tag register is nonzero, then the adapter will not respond to reads of the ID port and it will ignore the commands in the D1 to D7 range.
D8 to DF	Test adapter *n*, where *n* is the last three bits of the command. If the adapter tag register is not equal to *n*, then go to the ID_WAIT state, else the function is nonoperational.
E0 to FE	Activate adapter, write the last five bits of the command into the low five bits of the Address Configuration register that controls the I/O base address, and return to the ID_WAIT state.
FF	Activate the adapter at the preconfigured I/O base address and return to the ID_WAIT state.

In the ID_CMD state, I/O reads to the ID port are treated as a contention test. During a contention test, the adapter drives bit 15 of the EEPROM Data register out onto bit 0 of the host data bus, using an open drain driver. If there are multiple adapters in the system and one of them drives out a “1” and the other drives out a “0,” then both adapters and the host system will read a “0” on bit 0 of the host data bus. At the end of the read cycle the adapter samples the data on bit 0 of the host data bus. If the data does not match what was driven out, then the IDS has a contention failure and returns to the ID_WAIT state. The EEPROM Data register is shifted left one bit. Meanwhile, the host gets one bit of configuration data in bit 0 of the AX register with each read cycle.

This mechanism (ID_WAIT state and ID_CMD state) continues to function even after the adapter is activated. This means the adapter will respond to the software in the same way after either a cold or a warm boot.

Use the following algorithm to activate a particular adapter, assuming multiple randomly configured adapters exist in the host system.

1. Power-up the system.

All adapters autoinitialize and then enter the ID_WAIT state. In this state the adapter only responds to writes to the ID port. The ID port is the last I/O port in the range of 100h to 1F0h that has had a zero written to it.

2. **Write two 0 bytes and then the ID sequence to the ID port. All adapters enter the ID_CMD state.**
3. **Select EEPROM data to contend on by writing an EEPROM read command to the ID port.**
4. **Read the ID port 16 times. Any adapter that gets a contention failure will return to the ID_WAIT state.**
5. **Repeat steps 3 and 4 until contention criteria (station address, I/O base address, ROM base address) have been met and only a single adapter is left in the ID_CMD state.**
6. **Either tag the adapter with the Set Tag command or activate the adapter with the Activate command.**
7. **Repeat steps 1 through 6 until the desired adapter is found and activated.**
8. **Enable IRQ drivers by setting the ENA (enable) bit (bit 0 of the Configuration Control register).**

ISA Configuration Overview (3C509B)

You can configure 3C509B ISA adapters in one of three ways:

- ISA contention
- ISA Plug and Play
- EISA configuration

In an ISA bus system, there are two independent ways of locating and activating an adapter. The first is a contention scheme compatible with the 3C509 ISA adapter, which is called the Classic scheme. The second is the new Plug and Play (PnP) scheme. Host software can use either or both of these schemes when the corresponding mechanism is enabled in the ISA Activation Select field of the Internal Configuration register. For optimal results, the host software first attempts to activate the adapter using the PnP scheme. If the 3C509B does not respond, use the Classic scheme to activate it.

Both schemes involve contending down to a single adapter and then enabling it at a specific I/O base address (which may come from the EEPROM Address Configuration word). Additional configuration can then be done by writing to configuration registers in Window 0 and elsewhere.

The PnP scheme has the advantage that it can detect and avoid conflicts with other PnP devices. It also supports a better method of detecting whether an I/O address is in use or not, which should work whether the motherboard has pull-ups on the ISA bus or not.

In an EISA slot the adapter behaves the same as the 3C509 ISA adapter. It can be placed in EISA mode using the diagnostic/configuration program, which causes it to respond as an EISA adapter.

ISA Plug and Play Activation Mechanism (3C509B)

The adapter completely supports isolation, resource allocation, and activation by the Plug and Play mechanisms. The ISA Plug and Play specification describes the algorithms that should be used by host software.

The ISA Plug and Play activation mechanism is enabled when the I/O base is set to an ISA base address (200h–3E0h) and the ISA Activation Select bit (Internal Configuration register) is set to either "Both mechanisms enabled" or "ISA Plug and Play only."

It is recommended that driver software discover the presence or absence of a Plug and Play registry and then proceed accordingly:

Registry is detected. 3C509B adapters can be found by searching the registry (3C509 and 3C579 adapters will not be in the registry). Adapter configuration (I/O base, etc.) can be obtained from the registry. The adapter will already have been placed in the active state by the Plug and Play system software.

No registry. Either use Plug and Play mechanisms to resolve any resource conflicts with other Plug and Play adapters, or use the ISA activation algorithm described in the previous section.

NOTE: *When the adapter is in its initial state (3Com ID_WAIT state or the equivalent Plug and Play Wait4Key state), the Plug and Play Initiation Key mechanism has priority over the 3Com ID sequence; that is, an in-progress ID sequence will be aborted by the adapter if any write is detected to the Plug and Play Address port, and if Plug and Play initiation is in progress (either the Initiation Key is being received, or the adapter is no longer in the Wait4Key state), an ID sequence from the host will be ignored.*

This adapter has three resources that may be set by the Plug and Play allocation mechanism: I/O base address, interrupt request level (IRQ), and memory base address (optional boot ROM). When the Plug and Play registers corresponding to these resources are written to, the contents are also transferred into their respective Address Configuration and Resource Configuration fields. Invalid values written to those registers will cause undefined behavior; host software must write only values defined as valid in the Resource Data register.

All required ISA Plug and Play I/O ports and registers are supported. Refer to the Plug and Play specification for the functional definitions of the I/O ports and registers listed in Tables 7-1 and 7-2. Any Plug and Play registers that are not shown in these two tables return zero when read.

Table 7-1. I/O Ports Used for Plug and Play Commands

Name	I/O Port	Access Level: Read	Write
AddressPort	0x279		X
WritePort	0xA79		X
ReadPort	0x203-0x3FF (relocatable)	X	

Table 7-2. Supported Plug and Play Registers

Name	Register	Access Level: Read	Write	Notes
Set RD_DATA Port	00h		X	
Serial Isolation	01h	X		EEPROM data, 9 bytes from offsets 18h–1Ch. (Order is low byte first, high byte second. Low bit of the byte is first.)
Config Control	02h		X	
Wake (Card Select Number)	03h		X	
Resource Data	04h	X		EEPROM data, offsets 18h–1Ch. (Order is low byte first, high byte second.)
Status	05h	X		
Card Select Number	06h	X	X	
Logical Device Number	07h	X	X	0 (single device)
Activate	30h	X	X	
I/O Range Check	31h	X	X	
Memory Base Address Hi	40h	X	X	Same as Address
Memory Base Address Lo	41h	X	X	Configuration register
Memory Control	42h	X		0 (fixed memory size)
I/O Base Address Hi	60h	X	X	Same as Address
I/O Base Address Lo	61h	X	X	Configuration register
Interrupt Request Level	70h	X	X	Same as Resource
Interrupt Request Type	71h	X	X	Configuration register. Type = 02h (edge-triggered, active high).

EISA Activation Mechanism (3C509, 3C509B, 3C579)

If the I/O base address is set to the EISA slot-specific addressing mode, then the EISA activation mechanism is selected. When EISA activation is enabled, the ISA Activation Select bit (Internal Configuration register) is ignored.

After automatic configuration, the adapter responds to I/O accesses in the range of zC80 to zC8F (where *z* is the slot number into which the adapter is plugged), where the Window 0 register set is always mapped. The EISA motherboard BIOS checks the Manufacturer ID and Product ID registers, overwrites the Address Configuration and Resource Configuration registers, and then sets the Enable bit in the Configuration Control register to enable the adapter. Once enabled, the adapter responds to I/O accesses in the range of x000 to x00F where the complete adapter register set is mapped, to read access to its boot PROM address range (if any) and to enable the IRQ drivers. If the adapter is not installed in an EISA machine and the I/O Base is set to an ISA base address (100h–3E0h), then the IDS state machine is still enabled (see the preceding section) and can be used instead.

PCMCIA Activation Mechanism (3C589, 3C589B)

The 3C589 and 3C589B hardware supports the PCMCIA-specified architecture, allowing PCMCIA Card Services basic management of adapter resources. The adapter 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 adapter 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 adapter, allocate resources such as the I/O base and interrupt request level, and enable the adapter's I/O interface. The driver should save the I/O base and IRQ values for subsequent adapter operation. See the PCMCIA 2.xx specification for initialization details and the programming interface to Card Services.

Once the adapter's I/O interface is enabled, the adapter'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.

PCMCIA Configuration Overview (3C589, 3C589B)

The PCMCIA adapter 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 adapter uses the dedicated resources of the slot where it is installed. Therefore, the configuration register bits that normally control how the adapter assigns resources no longer apply and are in general ignored. The resources of a PCMCIA adapter 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 adapter 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 adapter 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 7-2 and Figure 7-3.

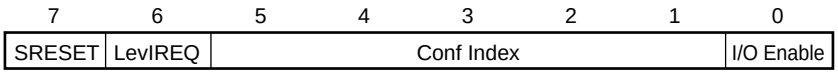


Figure 7-2. Configuration Option Register (3C589)

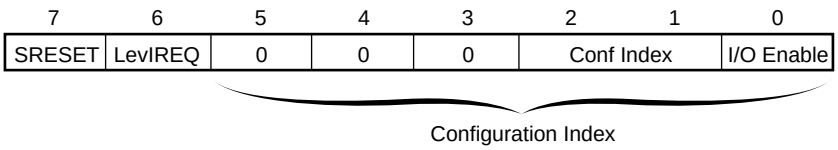


Figure 7-3. Configuration Option Register (3C589B)

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.

LevlREQ — 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
	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.

PCMCIA Card Configuration and Status Register

The Card Configuration and Status Register contains information about the card's condition. Refer to Figure 7-4 and Figure 7-5.

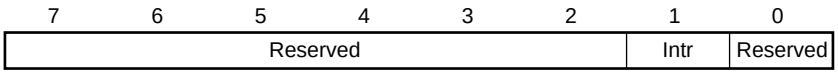


Figure 7-4. Configuration and Status Register (3C589)

Figure 7-5. 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 adapter 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.

MCA Programmable Option Select (POS) Registers (3C529)

POS registers are written by the system configuration program at boot-up. These registers are selected by the Card Setup signal (–CD SETUP) of the Micro Channel bus and may not be written at any other time. All of the bits are readable in Window 0. Refer to Figure 7-6, which shows two bytes containing the adapter ID.

Adapter ID Registers

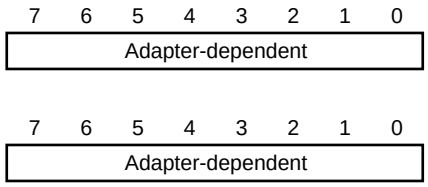


Figure 7-6. Adapter ID Registers

The Adapter ID is loaded from the EEPROM after a Channel Reset, a CCReset, a low-power reset, or an AISM Reset. The Adapter IDs are listed below. If the host tries to read the Adapter ID before the Autoinitialization state machine is complete, a value of 0000h is read. If the adapter is configured for test mode, the Adapter ID is forced to 61DB.

32-bit adapters:

10BASE2	ID = 627C
10BASE-T	ID = 627D
Test mode	ID = 61DB
TP or coax	ID = 62F6
TP only	ID = 62F7

Card Enable Register

CDEN: Card Enable. When this bit is reset to zero, the adapter does not respond to any host accesses except setup cycles. All DMA requests or interrupts are disabled if this bit is set to zero. It must be set to one by the system configuration program. In test mode, the CDEN enable function is masked so that the adapter behaves as if CDEN were always enabled. This bit is still read/write in test mode. Refer to Figure 7-7.

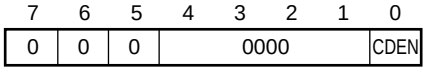


Figure 7-7. Card Enable Register

ROM Size and Base Register

ROM SIZE — ROM Size. Selects the window size of the boot PROM. The PROM may be 8 K, 16 K, or 32 K.

ROM BASE — ROM Base Address. Selects the base address for the boot PROM. The allowed combinations of the ROM Size and ROM Base are shown below. A Channel Reset resets the ROM Base to all zeros, which disables the boot PROM so it cannot be read until after the system configuration program has been run. Refer to Figure 7-8.

ROM Size	ROM Base	Address Window
XX	0000	Disable Boot PROM
00	0001	C2000h to C3FFFh
00	0010	C4000h to C5FFFh
00	0011	C6000h to C7FFFh
00	0100	C8000h to C9FFFh
00	0101	CA000h to CBFFFh
00	0110	CC000h to CDFFFh
00	0111	CE000h to CFFFFh
00	1000	D0000h to D1FFFh
00	1001	D2000h to D3FFFh
00	1010	D4000h to D5FFFh
00	1011	D6000h to D7FFFh
00	1100	D8000h to D9FFFh
00	1101	DA000h to DBFFFh
00	1110	DC000h to DDFFFh
00	1111	DE000h to DFFFFh
01	0001	C0000h to C3FFFh
01	001X	C4000h to C7FFFh
01	010X	C8000h to CBFFFh
01	011X	CC000h to CFFFFh
01	100X	D0000h to D3FFFh
01	101X	D4000h to D7FFFh
01	110X	D8000h to DBFFFh
01	111X	DC000h to DFFFFh
10	0001	C0000h to C7FFFh
10	01XX	C8000h to CFFFFh
10	10XX	D0000h to D7FFFh
10	11XX	D8000h to DFFFFh

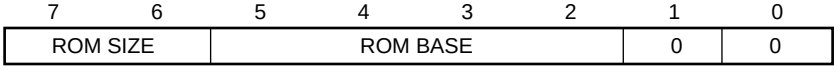


Figure 7-8. ROM Size and Base Register

I/O Base Register

XCVR: Transceiver Type Select. Read/write.
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 = 10BASE2 (BNC) transceiver enabled. The software driver must issue a Start Internal Transceiver command to the Command register to start the DC-DC converter.

IOBASE: I/O Base Address Select. These six bits determine the I/O Base address of the adapter. These six bits are compared to the last six bits of the I/O address range, A (15:10). The next six bits are fixed at 100000 and the final four bits select a particular register. There are 64 possible address ranges from 0200h to FE00h. The Forced Configuration option forces a base address of 0200h. Refer to Figure 7-9.

0110 1	000 XXXX XXXX

Fig 6-16. Acknowledge Interrupt

Figure 7-9. I/O Base Register

IRQ Register

IRQ: Interrupt Request Level. These four bits select the interrupt level to the Micro Channel host. Values of 3, 5, 7, 9, 10, 11, 12, or 15 enable the corresponding interrupt driver. All other values disable IRQ line drivers. Refer to Figure 7-10.

7	6	5	4	3	2	1	0
1	1	0	0	IRQ			

Figure 7-10. IRQ Register

Window 0 Configuration Registers

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

ISA/EISA/PCMCIA (3C509, 3C509B, 3C579, 3C589, 3C589B)

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 7-11.

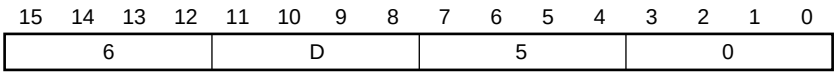


Figure 7-11. Manufacturer ID Register

Product ID Register (3C509, 3C509B, 3C579, 3C589, 3C589B) (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 7-12.

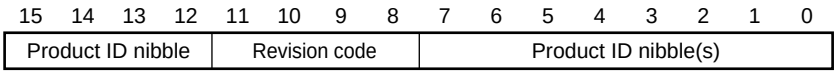


Figure 7-12. Product ID Register

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 adapter revision code. Changes to this field do not require a new version of the EISA configuration file.

PCMCIA (3C589, 3C589B)

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.

Adapter ID Register (3C529) (Read Only - Offset 2)

The automatic configuration logic loads this register. This adapter ID is the same as the Adapter ID described earlier in this chapter. This register is writable in Window 0 but is only to be used for test purposes. Refer to Figure 7-13.

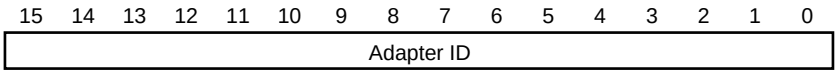


Figure 7-13. Adapter ID Register (MCA)

Configuration Control Register (3C509, 3C509B, 3C579, 3C589, 3C589B) (Read/Write - Offset 4)

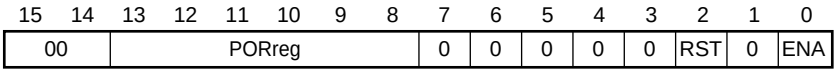


Figure 7-14. Configuration Control Register (3C509, 3C579, 3C589)

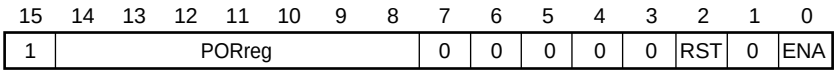


Figure 7-15. Configuration Control Register (3C509B, 3C589B)

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

Bit 14
0 = PCMCIA (3C589B) bus interface.
1 = ISA (3C509B) 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 3C509, 3C579, 3C589 (see Figure 7-14):
0 = Use external encoder/decoder.
1 = Use internal encoder/decoder.

For 3C509B and 3C589B (see Figure 7-15):
0 = Use external VCO.
1 = Use internal (on-chip) VCO.

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

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

To disable an adapter in EISA mode, you must first clear the EISA configuration register at zC84. Then bit 0 must be set to zero.

1 = Adapter enabled. In ISA mode, the adapter driver must set this bit. In EISA mode, this bit will be set by the BIOS. In a PCMCIA adapter, this bit is ignored.

Configuration Control Register (3C529) (Read/Write - Offset 4)

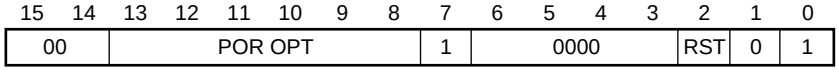


Figure 7-16. Configuration Control Register (MCA)

PORopt: Power-on reset options (see Figure 7-16). The bit descriptions are as follows:

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 — 0 = Use external encoder/decoder.
1 = Use internal encoder/decoder.

Bit 2 — 1 = Reset adapter. This bit resets everything, including the POS registers, which means the adapter will not be visible after asserting RST.

Bit 0 — Always 1.

Address Configuration Register (3C509, 3C509B, 3C579, 3C589, 3C589B) (Read/Write - Offset 6)

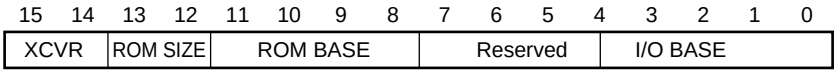


Figure 7-17. Address Configuration Register (3C509, 3C579, 3C589)

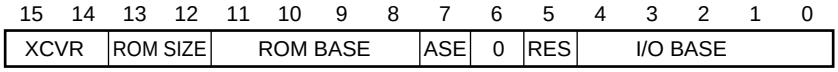


Figure 7-18. Address Configuration Register (3C509B, 3C589B)

Bit 7 contains software configuration information only; that is, it does not control adapter 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 7-17 and Figure 7-18.

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.

ISA/EISA (3C509, 3C509B, 3C579)

ROM SIZE: Boot PROM window size for 3C509, 3C579.

ROM SIZE: Boot PROM size for 3C509B.

Bits 13–12 ROM SIZE. Specifies the size of the ROM installed, if any. See below. ROMs larger than 16 K will still use only a 16 K window in host memory and must be paged using the ROM Page bits in the ROM Control register.

ROM BASE: Boot PROM base address select.

ROM Size = 8 K	ROM Base	Effective Address Window
00	0000	Disable boot PROM
00	0001	C2000h to C3FFFh
00	0010	C4000h to C5FFFh
00	0011	C6000h to C7FFFh
00	0100	C8000h to C9FFFh
00	0101	CA000h to CBFFFh
00	0110	CC000h to CDFFFh
00	0111	CE000h to CFFFFh
00	1000	D0000h to D1FFFh
00	1001	D2000h to D3FFFh
00	1010	D4000h to D5FFFh
00	1011	D6000h to D7FFFh
00	1100	D8000h to D9FFFh
00	1101	DA000h to DBFFFh
00	1110	DC000h to DDFFFh
00	1111	DE000h to DFFFFh

ROM Size = 16 K	ROM Base	Effective Address Window
01	0000	Disable boot PROM
01	0001	C0000h to C3FFFh
01	001x	C4000h to C7FFFh
01	010x	C8000h to CBFFFh
01	011x	CC000h to CFFFFh
01	100x	D0000h to D3FFFh
01	101x	D4000h to D7FFFh
01	110x	D8000h to DBFFFh
01	111x	DC000h to DFFFFh

ROM Size = 32 K	ROM Base	Effective Address Window
	ISA/ EISA (3C509, 3C579)	
10	0000	Disable boot PROM
10	0001	C0000h to C7FFFh
10	01xx	C8000h to CFFFFh
10	10xx	D0000h to D7FFFh
10	11xx	D8000h to DFFFFh
	ISA (3C509B)	
10	0000	Disable boot ROM
10	0001	C0000h to C3FFFh
10	001x	C4000h to C7FFFh
10	010x	C8000h to CBFFFh
10	011x	CC000h to CFFFFh
10	100x	D0000h to C3FFFh
10	101x	D4000h to D7FFFh
10	110x	D8000h to DBFFFh
10	111x	DC000h to DFFFFh

ROM Size = 64 K	ROM Base	Effective Address Window
	ISA/ EISA (3C509, 3C579)	
11	0000	Disable boot PROM
11	xxxx	Undefined - reserved
	ISA (3C509B)	
11	0000	Disable boot ROM
11	0001	C0000h to C3FFFh
11	001x	C4000h to C7FFFh
11	010x	C8000h to CBFFFh
11	011x	CC000h to CFFFFh
11	100x	D0000h to C3FFFh
11	101x	D4000h to D7FFFh
11	110x	D8000h to DBFFFh
11	111x	DC000h to DFFFFh

ISA/PCMCIA (3C509B, 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 adapter. 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 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.

Address Configuration Register (3C529) (Read/Write - Offset 6)

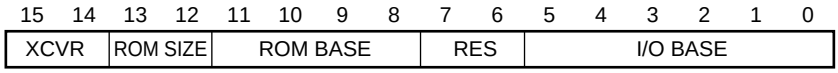


Figure 7-19. Address Configuration Register (3C529)

The bits in this register are read only and identical to the data stored in the POS bits with the same names. This allows the driver to know the configuration of the adapter without having to read the POS registers. Refer to Figure 7-19.

XCVR: Transceiver Type Select. Read only. Set in POS.
- 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.

I/O Base: I/O Base Address Select. These six bits determine the I/O Base address of the adapter. These six bits are compared to the last six bits of the I/O address range, A (15:10). The next six bits are fixed at 100000 and the final four bits select a particular register. There are 64 possible address ranges from 0200h to FE00h. The Test Mode option forces a base address of 0200h. These bits are set in POS and are read only in this register.

ROM SIZE: Boot PROM window size. The bits in this register are read only.

ROM BASE: Boot PROM base address select. The bits in this register are read only.

Resource Configuration Register (3C509, 3C579, 3C509B, 3C589, 3C589B) (Read/Write - Offset 8)

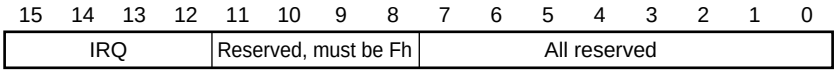


Figure 7-20. Resource Configuration Register (3C509, 3C579, 3C589, 3C589B)

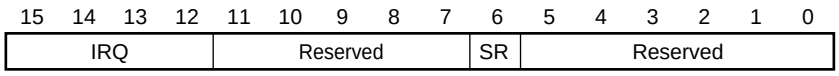


Figure 7-21. Resource Configuration Register (3C509B)

The automatic configuration logic loads this register from the EEPROM, offset 0x09 (except PCMCIA, in which the driver must load the register). 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.

ISA (3C509B)

Bit 6 Synchronous Ready. Normal I/O cycles if clear. If set, the adapter will assert I/OCHRDY on every I/O access. This may allow the board to work in a noncompliant bus that would otherwise not be supported (for example, an ISA bus running faster than spec). This bit defaults to 0 on power-up, but is overwritten from the EEPROM.

Resource Configuration Register (3C529) (Read/Write - Offset 8)

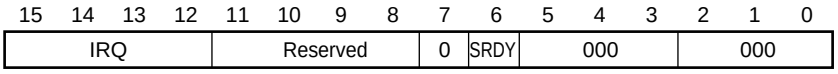


Figure 7-22. Resource Configuration Register (3C529)

The automatic configuration logic loads this register. Refer to Figure 7-22.

SRDY: Synchronous Ready. Setting this bit to one forces the adapter to do Synchronous Ready I/O cycles (300 ns) instead of default cycles (200 ns).

IRQ: Interrupt Request Level (3:0). These four bits select the interrupt level to the Micro Channel host. Values of 3, 5, 7, 9, 10, 11, 12, or 15 enable the corresponding interrupt driver. All other values disable IRQ line drivers. The bits in this register are read only and are set in POS register 5.

EEPROM Command Register (Read/Write - Offset A)

ISA/EISA/PCMCIA (3C509, 3C509B, 3C579, 3C589, 3C589B)

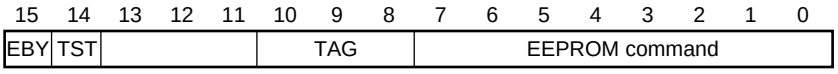


Figure 7-23. EEPROM Command Register (3C509, 3C509B, 3C579, 3C589, 3C589B)

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

MCA (3C529)

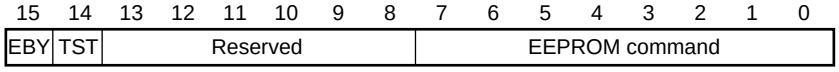


Figure 7-24. EEPROM Command Register (3C529)

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.

NOTE: *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.*

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

NOTE: *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.*

EEPROM Data Register (Read/Write - Offset C)

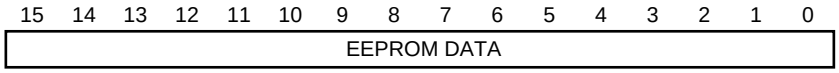


Figure 7-25. EEPROM Data Register

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 7-25.

ISA/EISA (3C509, 3C509B, 3C579)

The configuration and driver software must leave the Product ID in this register after using the EEPROM. Otherwise, the EISA system configuration software will not be able to identify the adapter after a warm boot.

Command Register (Read/Write - Offset E)

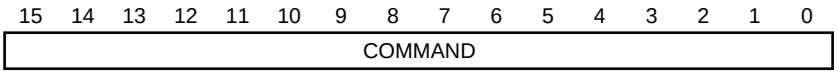


Figure 7-26. Command Register

The only valid commands in Window 0 are Window commands to select another window. Refer to Figure 7-26.

Window 3 Configuration Registers

ISA/PCMCIA (3C509B, 3C589B)

Function: Additional setup information.
Location: Window 3/Port 00h
Type: Read/write
Size: 16/32 bits

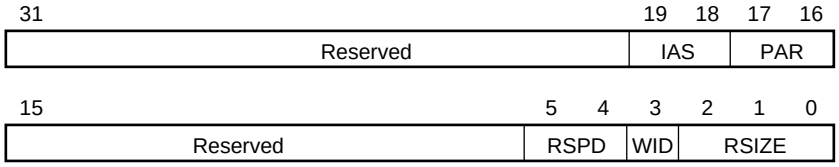


Figure 7-27. Internal Configuration Register

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 7-27.

Bit Description:

Bits 31–20 Reserved.

Bits 19–18 ISA ACTIVATION SELECT (3C509B only). Specifies which of the two ISA adapter activation mechanisms (Contention or Plug and Play) is enabled. When both mechanisms are enabled, host software may use either one to activate the adapter.
00b = Both mechanisms enabled (default)
01b = ISA contention only
10b = ISA Plug and Play only
11b = Both mechanisms enabled

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 adapter’s environment. An 8 KB adapter 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 clocks 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

ISA/EISA Test Mode and “Bad” Configuration Recovery

It is the user’s responsibility (with help from the configuration program) to avoid configuring the boot PROM on the adapter in such a way that the system is not able to boot. If this does occur, you can manually jump the Test Via using a #2 pencil. The Test Via forces the adapter into test mode, which disables the boot PROM so that the adapter configuration program can be run. You must cover the designated area thoroughly with the pencil mark.

Test mode forces the adapter not to perform the Automatic Initialization sequence, which means the EEPROM is not read and the adapter is left in the following configuration:

- Address Configuration register = 0000h
- Resource Configuration register = 0000h
- Product ID register = 0000h

Test mode also shortens the ID sequence to 8 bytes (first is still FFh and last is 69h) and forces it to the active state (that is, the adapter is active and will respond to I/O cycles at base address 0200h even without going through the ID sequence).

After you are done, thoroughly erase the pencil mark.

EEPROM Data Structure

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	3C5X9 Product ID	3C509 and 3C509B: 9150 3C509-TP and 3C509B-TP: 9050 3C509-COMBO, 3C509B-COMBO: 9450 3C509-TPO and 3C509B-TPO: 9550 3C529: 627C 3C529-TP: 627D 3C579: 9350 3C579-TP: 9250 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

ISA, MCA, EISA, PCMCIA (3C509, 3C529, 3C579, 3C589)

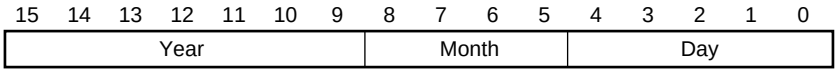
10 to 3F	Network Management Data	XXXX

ISA/PCMCIA (3C509B, 3C589B)

10	Capabilities Word	2083 for ISA, 2082 for PCMCIA
11	Reserved	0000
12	Internal Configuration Word 0	0000 for ISA/PCMCIA
13	Internal Configuration Word 1	0000 for ISA/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.)

“X” represents a value that may vary from adapter to adapter.

The manufacturing date format is shown in Figure 7-28.



- 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.

Figure 7-28. Manufacturing Date Format

MCA (3C529)

The Adapter ID is determined by the board type:

32-bit coax: 627C
32-bit twisted pair: 627D

EEPROM Data Structure, Offset 0Dh (Software Information)

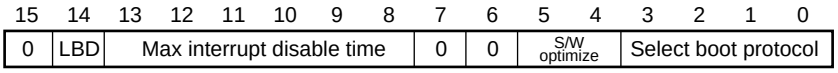


Figure 7-29. EEPROM Data Structure (Software Information)

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 7-29.

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, or when the 3C509 adapter is connected to certain pre-10BASE-T hubs.

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.

ISA (3C509B)

Bits 3–0 Select Boot Protocol. Specifies which one of the several protocols that may exist in the boot ROM is to be used during remote program loading.

EEPROM Data Structure, Offset 0Eh (Compatibility Word)

The EEPROM data structure, offset 0Eh (Compatibility Word) is shown in Figure 7-30.

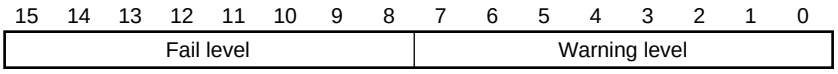


Figure 7-30. EEPROM Data Structure (Compatibility Word)

Bits 15–8 Fail Level. Indicates the hardware revision level as it pertains to software compatibility. For 3C509, 3C509B, 3C529, 3C579, and 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 3C509, 3C509B, 3C529, 3C579, and 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 adapter, 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 (3C509B, 3C589B)

The EEPROM data structure, offset 10h (Capabilities Word for ISA and PCMCIA [3C509B and 3C589B]), specifies the capabilities of this adapter:

Bit 0	Specifies whether the adapter supports Plug and Play. Set for ISA and PCMCIA (3C509B and 3C589B) only.
Bit 1	Always one for 3C509B and 3C589B.
Bits 2–6	Always zero for 3C509B and 3C589B.
Bit 7	CRC PASS THRU. Specifies whether the adapter supports CRC pass-through or not. Always set for 3C509B.
Bits 8–9	Always zero for 3C509B and 3C589B.
Bits 10–12	Reserved, read as zero.
Bit 13	POWER MANAGEMENT. Always set for 3C509B and 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 (3C509B, 3C589B)

The EEPROM data structure, offset 14h (Secondary Software Information for ISA and PCMCIA [3C509B and 3C589B]), is shown in Figure 7-31.

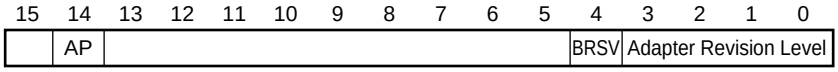


Figure 7-31. Secondary Software Information

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 adapter 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 — Adapter Revision Level.

00 = 3C509, 3C529, 3C579, 3C589.
01 = 3C509B, 3C589B.

EEPROM Data Structure, Offset 17h (Secondary Checksums)

ISA/PCMCIA (3C509B, 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 adapter diagnostic/configuration program. See the ISA Plug and Play specification for information on decoding this information.

PCMCIA-Specific Data Structures (3C589)

The Card Information Structure (CIS) used by the 3Com PCMCIA (3C589) adapter 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) adapter 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
```

Chapter 8
ISA/EISA/MCA/PCMCIA Bus Interfaces

Supported Slot Types and Cycle Types

Table 8-1 summarizes the slot types and cycle types supported by the EtherLink III adapters.

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
ISA	8-bit, 16-bit, 32-bit	8-bit, 16-bit, 32-bit (does not assert MEMCS 16)	Default timing as specified in the technical reference guide for the IBM® Personal Computer AT®. (The adapter does not assert zero wait state or de-assert IOCHRDY.)
EISA	8-bit, 16-bit, 32-bit	8-bit, 16-bit, 32-bit (in the same manner as an 8-bit ISA adapter)	Only default timing, as specified in the EISA specification version 3.10, section 2.11.4.1, ISA-compatible Timing Parameters.
MCA	8-bit, 16-bit, 32-bit[1]	8-bit and 16-bit as an 8-bit adapter (does not assert –CDDS16 or –CDDS32)[2]	Default timing as defined by the *Personal System/2 Hardware Interface Technical Reference Manual*.
PCMCIA	8-bit (ISA-like) and 16-bit (always asserts –IOIS 16)	8-bit transfers to Attribute Memory[3]	Only default timing, as specified in the PCMCIA release 2.01, PC Card Specification.

[1] The MCA adapter supports 32-bit cycles only to the PIO Data register at offset +00 (asserts –CDDS32 and –CDDS16). It supports all other I/O accesses as a 16-bit slave (asserts –CCDS 16 only). I/O cycles may optionally be selected as Synchronous Extended cycles by setting the SRDY bit.

[2] MCA memory cycles are done as Asynchronous Extended cycles.

[3] PCMCIA Configuration registers and Card Information Structure.

EISA/ISA Details

DC Characteristics – Pin Drive/Load Types

Type	3SH	3SL	OC	BI	IN	IN2	Units
V_{oh}	2.4	2.4	-	2.4			min volts
V_{ol}	0.4	0.4	0.4	0.4			max volts
I_{oh}	3.0	0.4	-	3.0			max mA
I_{ol}	24.0	5.0	24.0	24.0			max mA
V_{ih}				2.0	2.0	2.0	min volts
V_{il}				0.8	0.8	0.8	max volts
I_{ih}				10.0	10.0	20.0	max µA
I_{il}				10.0	10.0	20.0	max µA
I_{cap}	15	15	15	15	15	15	max pF

Board Edge Connector Pins

Pin No.	Pin Name	Description	Type	Notes
A2–9	SD7–0	Low byte data bus	BI	
A11	AEN	Address enable	IN	
A12–15	SA19–16	Latched address bus	IN	
A16–31	SA15–0	Latched address bus	IN2	2
B2	RESETDRV	Power-on reset	IN	
B4	IRQ9	Interrupt request 9	3SH	1
B12	SMEMR_	Memory read strobe	IN	
B13	IOW_	I/O write strobe	IN	
B14	IOR_	I/O read strobe	IN	
B21	IRQ7	Interrupt request 7	3SH	1
B23	IRQ5	Interrupt request 5	3SH	1
B25	IRQ3	Interrupt request 3	3SH	1
B27	T/C	Not used	IN	
C1	SBHE_	High-byte enable	IN	
C11–18	SD8–15	High-byte data bus	BI	
D2	IOCS16_	16-bit I/O select	OC	
D3	IRQ10	Interrupt request 10	3SH	1
D4	IRQ11	Interrupt request 11	3SH	1
D5	IRQ12	Interrupt request 12	3SH	1
D6	IRQ15	Interrupt request 15	3SH	1
D10	DACK5_	Not used	IN	
D11	DRQ5	Not used	3SH	
D12	DACK6_	Not used	IN	
D13	DRQ6	Not used	3SH	
D14	DACK7_	Not used	IN	
D15	DRQ7	Not used	3SH	
B1, B10, B31, D18	GND			
B3, B29, D16		+5 volts @ .4 A max.		3
B9		+12 volts @ .5 A max.		

NOTES:

1. *Although both the IBM technical reference guide and the EISA specification specify an open collector driver on the IRQ signals, you cannot assume there is a reasonable pullup resistor on these lines in all ISA machines. The 3C579 adapter uses tri-state drivers on these lines and drives the selected IRQ signal high (similar to 3Com's EtherLink16 adapter). This means that the 3C579 adapter does not support shared interrupts.*
2. *SA15–0 are the only bus interface signals that go to more than one IC pin on the 3C579 adapter. They are connected to both the 3C579 ASIC and the boot PROM address pins.*
3. *Max +12V current standby 0 mA*
   10BASE-T transceiver on 0 mA
   10BASE2 transceiver on 300 mA
   External transceiver on 500 mA

MCA Details

DC Characteristics

Signal Group	Driver Type	Sinking Capacity	Maximum Capacitive Loading
1	Tri-state	24 mA	20 pF
2	Tri-state	24 mA	20 pF
3	Totem pole	6 mA	20 pF
4	Bus driver	24 mA	20 pF
5	Open collector	24 mA	15 pF
6	Open collector	24 mA	20 pF
7	Clock driver	24 mA	15 pF
8	Totem pole or tri-state	6 mA	50 pF

NOTE: *Maximum loading current is 1.6 mA per channel connector, except signal group 5. The maximum loading current of group 5 is 1.0 mA per channel connector.*

Board Edge Connector Pins

Signal Name	I/O	Driver Type
A (23:0)	I	1
_ADL	I	1
_BE(3:0)	I	1
_CDDS32 (n)	O	3
CD CHRDY (n)	O	3
_CDDS16 (n)	O	3
_CDSETUP (n)	I	8
_CDSFDBK (n)	O	3
CHRESET	I	4
_CMD	I	1
D (31:0)	I/O	2
_IRQ (15,12:9,7,5,3)	O	6
M/_I/O	I	1
MADE24	I	1
_S (1:0)	I	1
_SBHE	I	1

NOTES:

1. *Max +5V current* *400 mA*
2. *Max +12V current standby* *0 mA*
   10BASE-T transceiver on *0 mA*
   10BASE2 transceiver on *300 mA*
   External transceiver on *500 mA, determined by external transceiver*

PCMCIA Details

Refer to Tables 8-2 and 8-3 for PCMCIA pin assignments. Notes for both tables appear after Table 8-3.

Table 8-2. PCMCIA Pin Assignments (Pins 1 Through 34)

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		
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		

Table 8-3. PCMCIA Pin Assignments (Pins 35 Through 68)

Pin	Signal	I/O Type	Function	Polarity	Notes
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				
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.*

Chapter 9
External Configuration Options

The EtherLink III adapter'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

Listed below are the I/O pins available to boundary scan and their functionality with respect to the ASIC.

ISA/EISA Only

Inputs	Outputs
SBHE_	DRQ[7:5]
AENA	EEDataO
IOR_	EEClockO
IOW_	IOCS16_
SMRD_	LEDDrvO_
SAIn[19:0]	IRQ[15, 12:9, 7, 5, 3]
EEPROMDIn	
SData[15:0]	

MCA Only

Inputs	Outputs
_SBHE	EEDataO
_ADL	EEClockO
_S [1:0]	LEDDrvO_
_CMD	
_CDSETUP	
MA [23:0]	
EEDataIn	
MData [31:0]	
MADE24	
_BE [3:0]	
CDCHRDY	
_CDDS16	
IRQ [15,12–9,7,5,3]	
_CDSFDBK	
_CDDS32	
M/_IO	

PCMCIA Only

Boundary scan is not supported.

Forced Configuration

ISA/EISA Only

The EEPROMDIn pin is sampled at the falling edge of ResetPinIn. The ASIC is configured with an I/O base address of 200h, the boot PROM is disabled, and the internal 10BASE-T transceiver enabled (independently of the availability of this resource). This option is exercised at board-level test, and can be used in extreme cases by customers if the 3C579 ASIC is incorrectly configured in a machine.

MCA Only

The EEPROMDIn pin is sampled at the falling edge of ResetPinIn. The ASIC is configured with an I/O base address of 200h, the boot PROM is disabled, the Adapter ID is 61DB, and the internal 10BASE-T transceiver enabled (independently of the availability of this resource). This option is exercised at board-level test, and can be used in extreme cases by customers if the 3C529 ASIC is incorrectly configured in a computer.

PCMCIA Only

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.

NOTE: *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.

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

PCMCIA Only

Only Receive and Transmit physical accesses are supported.

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)

PCMCIA Only

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.

Chapter 10
Adapter Differences

This chapter contains topics that explain differences between the various types of EtherLink III adapters. The first section discusses the differences among the 3C509, 3C529, 3C579, 3C589, and 3C589B adapters. The second section addresses the changes that occur in the current version (3C509B and 3C589B) of Parallel Tasking EtherLink III ISA and PCMCIA adapters. Some changes made to the MCA (3C529) adapter also apply to the second-generation ISA and PCMCIA adapters (3C509B and 3C589B). These changes are discussed in the final section.

Differences Among EtherLink III Adapters

The following sections contain MCA-specific considerations in relation to the other adapters, as well as PCMCIA specification information.

MCA Adapters

POS Registers

The 3C529 adapter supports the POS registers of the Micro Channel architecture. Several Window 0 register bits that are loaded from the on-board configuration EEPROM or are read or written on the 3C509/3C579 are read-only in Window 0 for the 3C529 adapter. These register bits are read/write in the POS registers of the 3C529. The following register bits are loaded from the EEPROM on the 3C509/3C579 but are POS bits in the 3C529: I/O BASE, ROM SIZE, ROM BASE, XCVR, and IRQ.

ISA Activation Mechanism Removed

The ID Sequence state machine used to configure the base I/O address on the 3C509 is removed in the 3C529. The Micro Channel architecture setup utility resolves I/O base address conflicts.

Adapter Enable Bit

The Configuration Control register bit 0, the Enable Adapter bit, is always a one in the 3C529. Writing to this bit position has no effect.

Adapter ID

The 3C529 loads its Adapter ID from EEPROM offset 3. The 3C509/3C579/3C589/3C589B loads a Product ID from offset 3. The IDs are unique for the two products, but their use is similar. The Adapter ID for the 3C529 references the correct adapter ADF file.

Reset Command Arguments

In the 3C529 the Global Reset, TX Reset, and RX Reset commands have an argument field that masks resets to various portions of the ASIC. Each bit masks a different portion of the ASIC. If the argument is all zeros, the command behaves the same as the 3C509/3C579/3C589/3C589B. Review Chapter 6, “Register Definitions,” for the command argument definitions.

Command in Progress

The 3C509/3C579/3C589/3C589B requires a poll of the Command-in-Progress bit when an RX Discard command is issued. The 3C529 always requires polling the Command-in-Progress bit for an RX Discard, but additionally the RX Reset and TX Reset commands should poll the Command-in-Progress bit.

TX Reclaim Threshold Command

This command, which does not exist in the 3C509/3C579/3C589/3C589B, is optional because the 3C529 default is identical to that of the 3C509/3C579/3C589/3C589B. The 3C509/3C579/3C589/3C589B requires the entire packet to be transmitted before the memory space in the transmit FIFO is reclaimed. The 3C529 adds a TX Reclaim mechanism. Once the Reclaim Threshold number of bytes has been transmitted, the Free Transmit bytes counter continues to increase as each byte is transmitted on the network. TX Status bit 1 is undefined for the 3C509/3C509B/3C579/3C589/3C589B; it becomes the TX Reclaim Error bit for the 3C529.

The TX Reclaim Threshold command is command field 11000, and the next three bits are the command argument. Refer to the Set TX Reclaim Threshold command in Chapter 6 for more details.

Adapter Failure Errors

Adapter Failure errors are reported when a transmit overrun or a receive underrun occurs. The 3C509/3C579/3C589/3C589B may incorrectly report a receive underrun Adapter Failure when no error has occurred. A driver should ignore this failure and recover by issuing an RX Reset. The 3C529 correctly reports transmit overruns and receive underruns.

RX Early Default and Disable

In the 3C509/3C579/3C589/3C589B the power-up default RX Early Threshold count is 2,032. In the 3C529 the default is 2,044. In either case, setting the RX Early Threshold greater than 1,792 (the largest packet that can be received) disables the function.

RX Discard Commands

In the 3C529, RX Discard commands issued before the packet is visible to the host are ignored. The 3C509/3C579/3C589/3C589B allows an RX Discard command to a packet that was not yet visible if it had begun to be received.

TX Start Threshold + 4

The 3C509/3C579/3C589/3C589B adds four to the Start Threshold command written by the host, which can be read in Window 5. This is removed in the 3C529 so that the Start Threshold is the same as that written to the host.

Synchronous Ready

The 3C529 adds a Synchronous Ready mode of operation as defined by the Micro Channel operational specifications. By setting the SRDY bit in Resource Configuration bit 6, the adapter asserts _CDCHRDY supporting Synchronous Ready cycles of 300 ns; otherwise, the adapter supports the default cycle of 200 ns.

TX PIO Write Command Followed Immediately by a TX Free Read Command

Fast Micro Channel architecture cycles introduce the possibility that a driver could write to the TX PIO register and immediately read the TX Free register. The TX Free register might act as if the write had not occurred. TX Free is updated by an internal arithmetic operation that occurs one clock after the write operation is completed. In a fast bus, an immediate read of TX Free may give a value that is not updated. If driver code reads TX Free near a TX PIO write, then as a precaution, the driver should subtract four from the result. It is possible to read zero in the TX Free register, so the driver must be prepared for a negative number.

Sixteen Hidden Receive Bytes

The 3C509/3C579/3C589/3C589B hides 16 bytes from the RX Byte count until the packet is completely received. Since the 3C529 does not hide these bytes, the RX Early Threshold may need to be adjusted by a driver. A secondary effect of this change is that programming a zero in the RX Early Threshold for the 3C509/3C579/3C589/3C589B behaves as if the threshold is set to 16 bytes. Programming a zero in the RX Early Threshold for the 3C529 behaves as if the threshold is set to eight. Any RX Early less than eight behaves as if set to eight.

RX Status Overrun

The 3C509/3C579/3C589/3C589B includes an RX Status Overrun indication at bit 12 of the FIFO Diagnostic register. The 3C529 eliminates this bit because it does not have an RX Status Overrun.

Continuous Statistics

In the 3C509/3C579/3C589/3C589B, if an Update Statistics interrupt is generated when the transmitter is in error (TX Status has TX Complete set), then it will not read the late collisions, carrier sense lost, or deferrals statistics. These may be incrementing at a high rate. To prevent this, 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. The 3C529 does not require the Read Zero mask for the interrupt. The 3C509/3C579/3C589/3C589B and 3C529 require that the Statistics Disable command be issued before the statistics are read.

RX Discard Anomaly

In the Receive routine for the 3C509/3C579/3C589/3C589B, after the driver issues an RX Discard command, the driver must poll the Command-in-Progress bit, and if the Complete bit is not set, the 3C509/3C579/3C589/3C589B should be checked for a correct byte count. If the byte count calculated from RX Free is not close to the RX Status byte count (they will not be exactly the same, since the adapter may be receiving, but they should be within 100 bytes), the driver should issue an RX Reset and restore the adapter to its operating state. The byte count calculated from RX Free is determined by the RX FIFO empty value of 2,044 bytes minus the value in RX Free. The 3C529 does not require this check, but leaving the code in the driver will not cause a problem.

PCMCIA Adapters

3Com PCMCIA adapters support the Card Information Structure (CIS), Configuration Option register, and Card Configuration and Status register defined in the PCMCIA specification.

Changes in ISA and PCMCIA Adapters

Significant changes have been made in the current (3C509B and 3C589B) version of the EtherLink III ISA and PCMCIA adapters.

ISA Adapter Changes

This section is limited to the changes in ISA (3C509B) adapters relative to the previous (3C509) version.

Plug and Play

The 3C509B now supports the Microsoft Plug and Play (PnP) specification. This allows the adapter resources to be automatically allocated by the operating system, much as current EISA/MCA systems do, to avoid conflicts with other adapters. Software can continue to use the old contention scheme introduced with the previous version of adapters or use the PnP method to find and configure boards. The PnP scheme includes a way to detect whether an I/O port is in use or not and whether there are pull-ups on the bus or not. Such detection allows for improved auto-configuration via the diagnostic/setup program.

ISA Adapter Configuration as EISA

The 3C509B can be configured so that it responds as an EISA adapter simply by being inserted in an EISA slot. The EISA mode supported in the previous (3C509) version of ISA adapters is still supported but is no longer required.

ISA Adapter in Higher-Speed Buses

The 3C509B works in buses where the 3C509 would not. The 3C509B supports higher-speed ISA buses in its native operation mode. It also adds an alternative Early Ready timing that should allow it to work in almost all cases, though by imposing a slight performance/CPU utilization penalty.

CMOS EEPROM Support

The 3C509B supports use of CMOS EEPROM for the boot ROM. Such boot ROMs work exactly the same as the previous version of boot ROMs, but can be updated if desired. This allows updatable boot ROMs to be built for the 3C509B at a much lower product cost than for the 3C509.

Changed ROM Support

The 3C509B changes the ROM support in a variety of ways: full decode of the ROM address allows the use of real 8 K or 16 K parts if desired (rather than 32 K parts with multiple copies of the 8 K or 16 K image), support for 64 K parts, and support for switching 16 K windows into any part larger than 16 K. It is no longer possible to use a 32 K part without window switching.

LED Indicator on TP Adapters

The LED on TP adapters is OFF at power-up. If the link beat is not enabled, the LED will be OFF, unless polarity swap has been detected, in which case the LED will flash.

ISA and PCMCIA Adapter Changes

This section summarizes the changes in ISA (3C509B) and PCMCIA (3C589B) adapters relative to the previous version (3C509 and 3C589, respectively). These changes may affect the functionality of the host software.

External RAM

RAM is now external rather than on-chip. The 3C509 and 3C589 had 4 K of on-chip RAM. The 3C509B and 3C589B now use external SRAM. The size of the SRAM can vary from 8 K to 32 K. The 3C509B and 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 3C509 and 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 3C509B and 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 3C509B and 3C589B support 8-bit accesses, which the 3C509 and 3C589 adapters did not. This will allow them to be used in 8-bit slots or in 8-bit designs (like handhelds), where 3C509 and 3C589 adapters could not be used.

CRC Pass-Through Support

The 3C509B supports CRC pass-through. A static configuration option allows CRCs to be received along with the packet data. A bit in the transmit preamble allows a per-packet inclusion of the CRC with the data. This allows bridges to perform CRC pass-through properly.

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.

Immediate Transmit Underruns Reported

The TX Status register in the 3C509B reports transmit underruns immediately, rather than waiting until the bad CRC has been transmitted. Software can therefore get immediate underrun information without reading the Ethernet Controller register, as it had to with the 3C509.

To make the above work properly in a backward-compatible way, the TX Reset command has a new bit defined in the argument field. The TX Reset command, with this bit set to zero, defers until transmitting is completed, allowing the bad CRC to finish being transmitted. Setting the bit to one causes a transmitter reset immediately, possibly before a guaranteed bad CRC is generated.

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 ISA, MCA, and PCMCIA Adapters

A few of the changes made in the MCA adapter (3C529) also apply to the ISA and PCMCIA adapters (3C509B and 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.

Appendix A
Driver Routine Flowcharts

This appendix contains process flowcharts. They help software developers write code that best incorporates the features of the 3C509, 3C509B, 3C529, 3C579, and 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 (ISA, EISA, or MCA).
- Check Slot (EISA or MCA) - 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 (ISA/EISA/MCA)

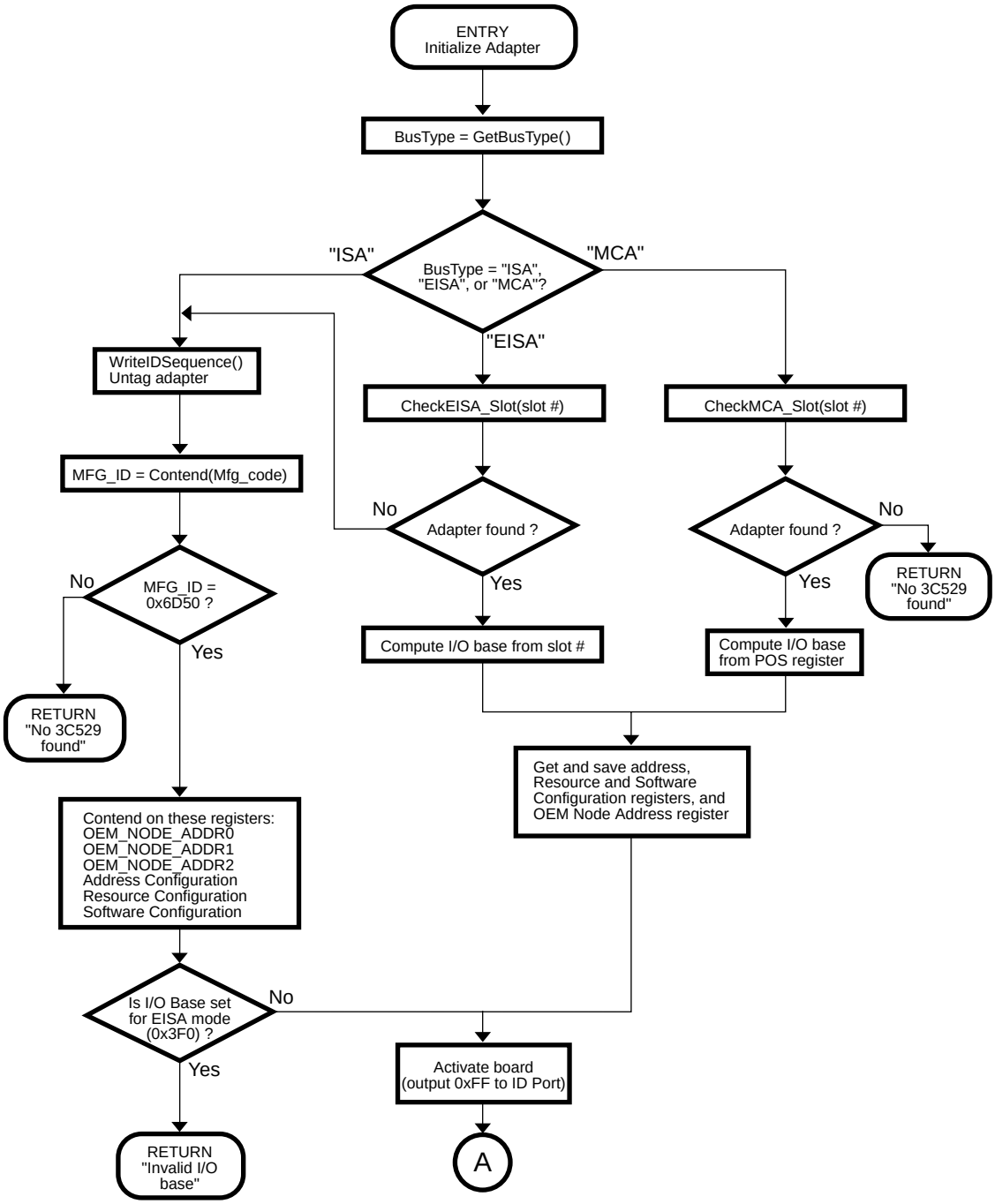


Figure A-1. Initialize Adapter (ISA/EISA/MCA)

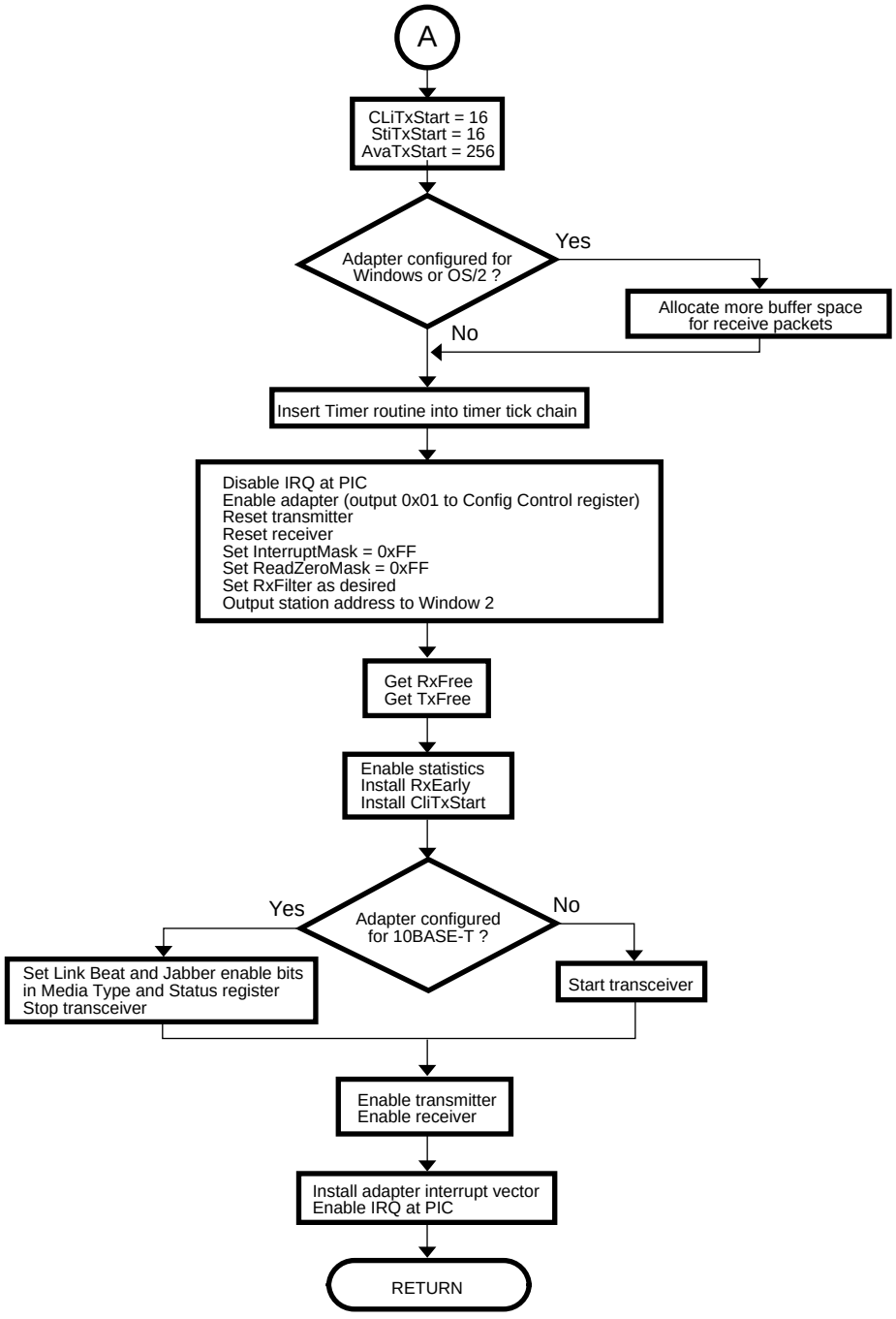


Figure A-2. Initialize Adapter (ISA/EISA/MCA) (continued)

Initialize Adapter (PCMCIA)

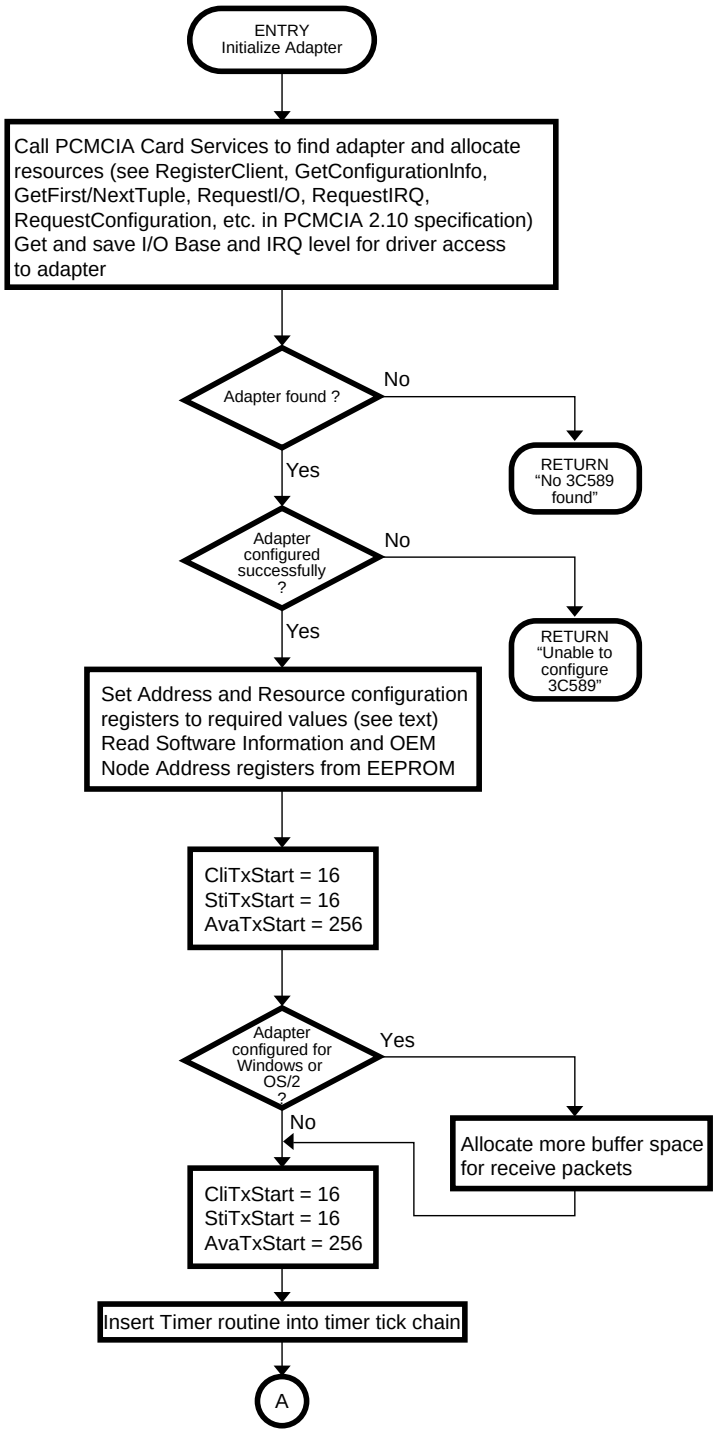


Figure A-3. Initialize Adapter (PCMCIA)

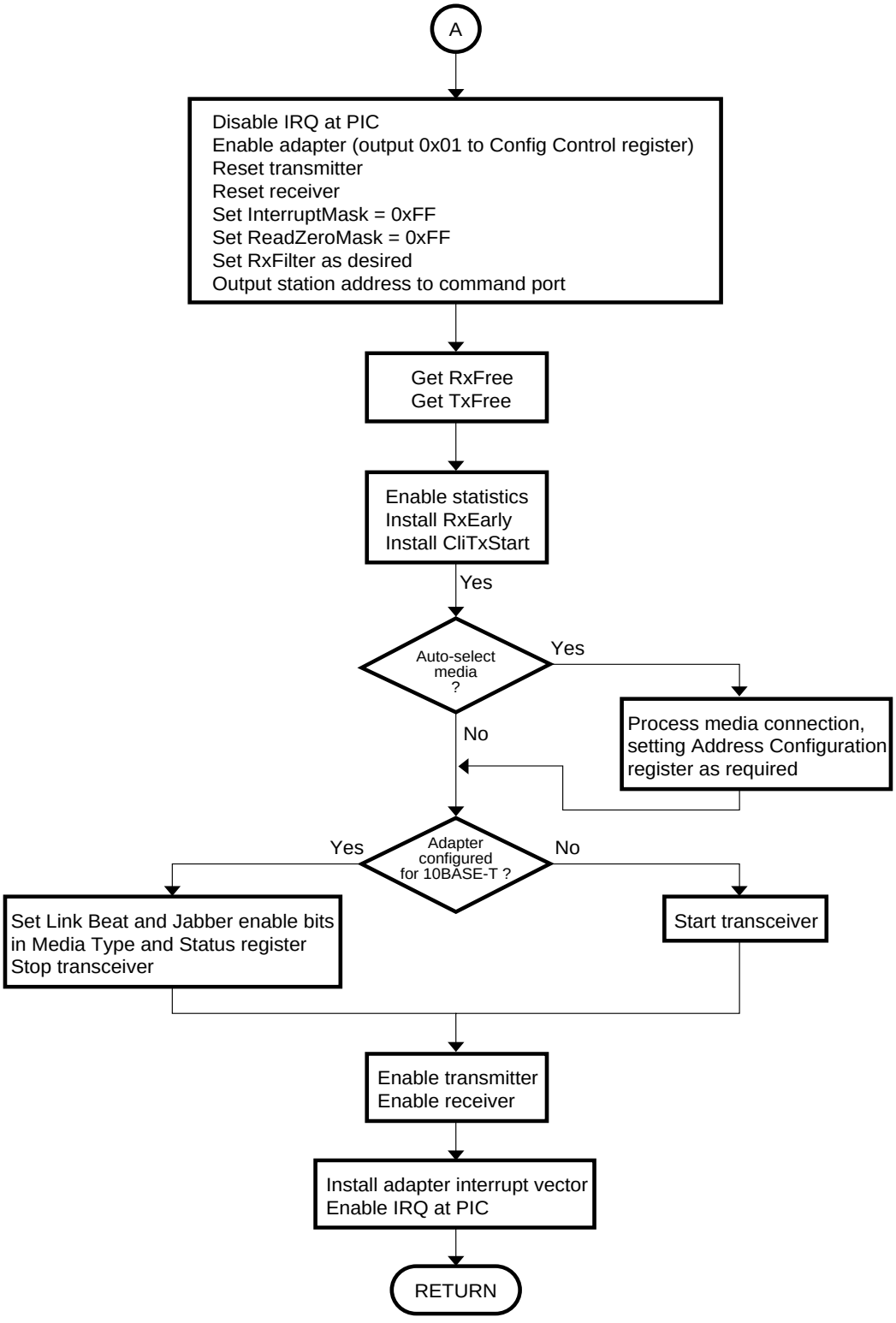


Figure A-4. Initialize Adapter (PCMCIA) (continued)

Get Bus Type

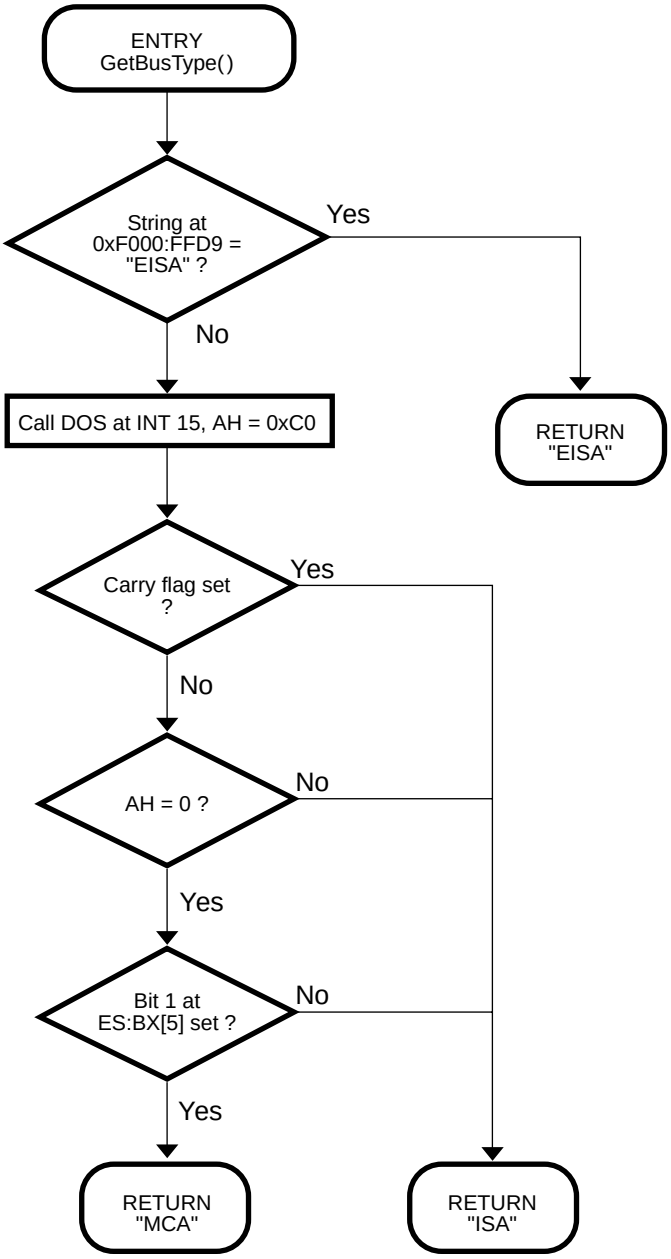


Figure A-5. Get Bus Type

Check Slot (EISA)

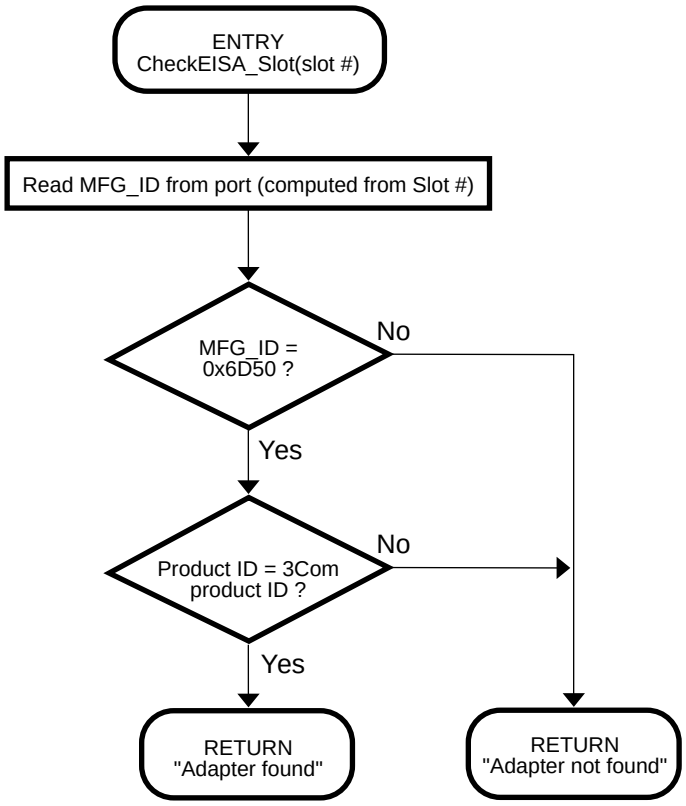


Figure A-6. Check Slot (EISA)

Check Slot (MCA)

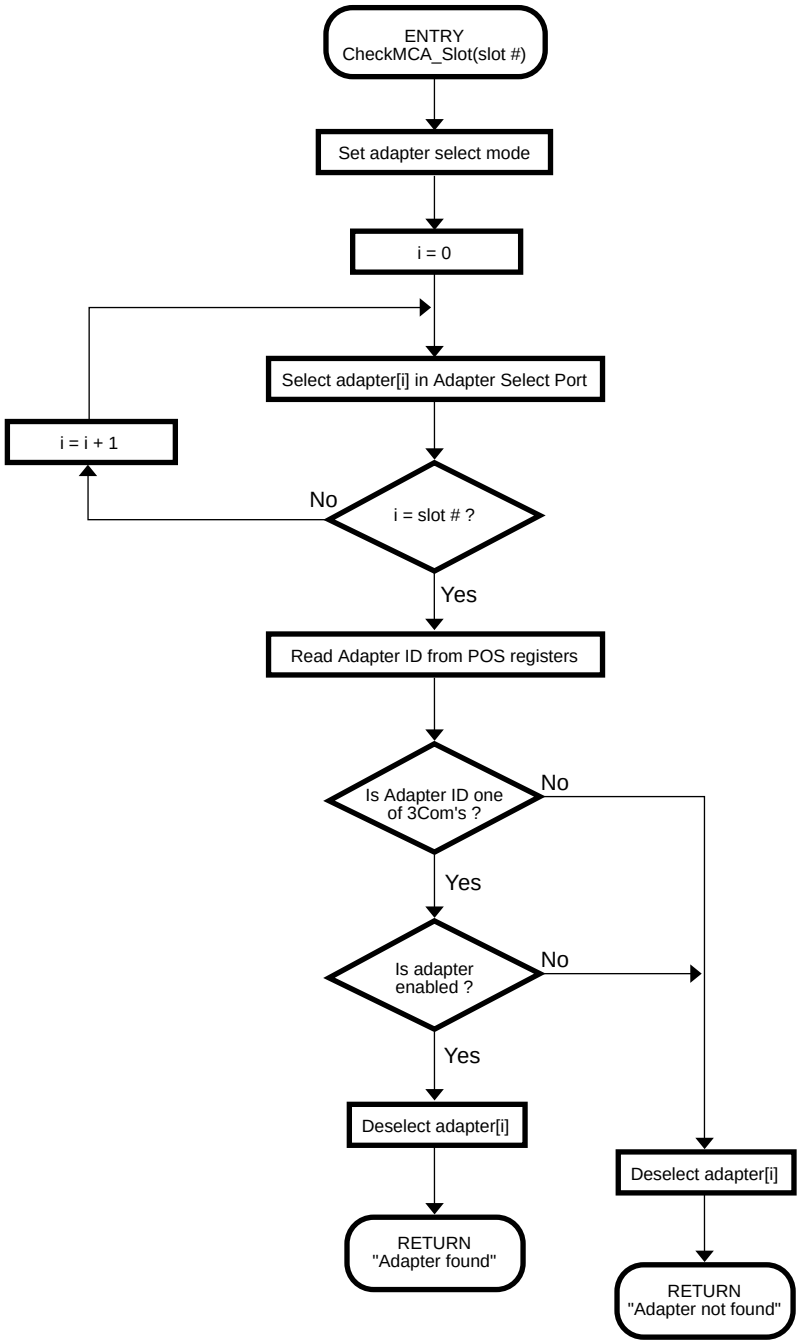


Figure A-7. Check Slot (MCA)

Write ID Sequence (ISA/EISA/MCA)

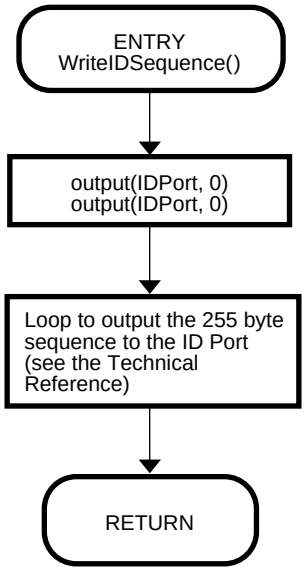


Figure A-8. Write ID Sequence (ISA/EISA/MCA)

Transmit Frame

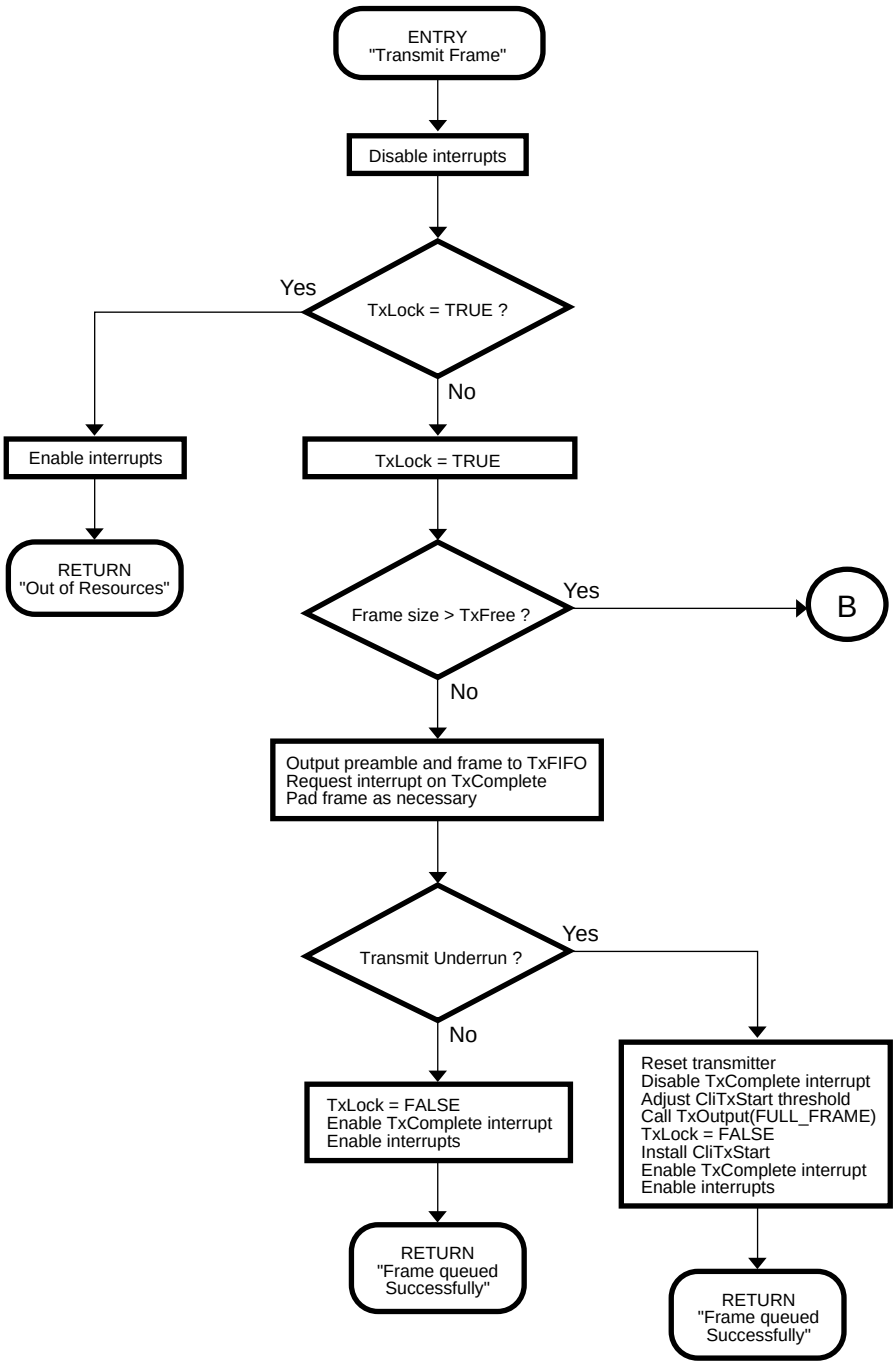


Figure A-9. Transmit Frame

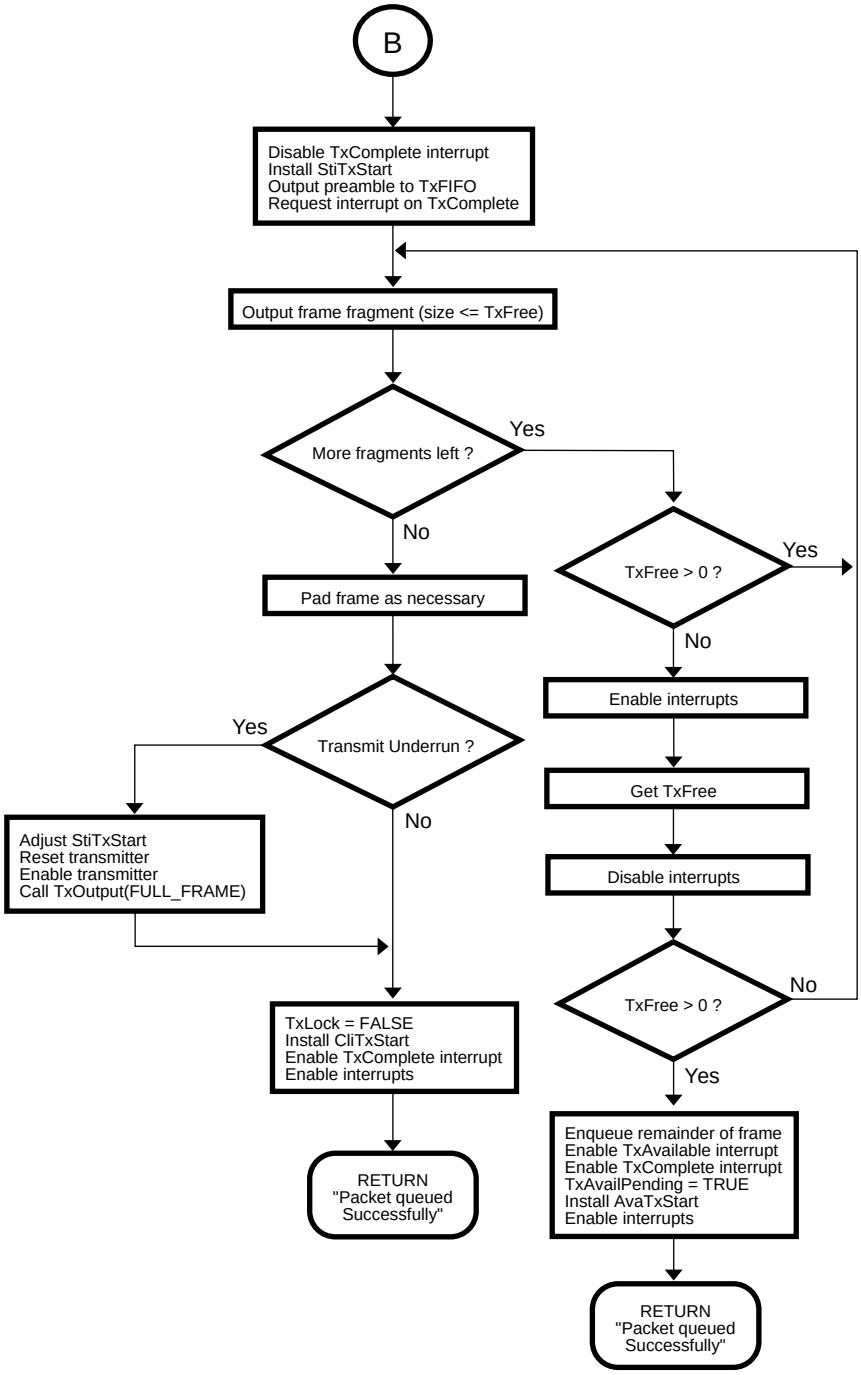


Figure A-10. Transmit Frame (continued)

Transmit Output

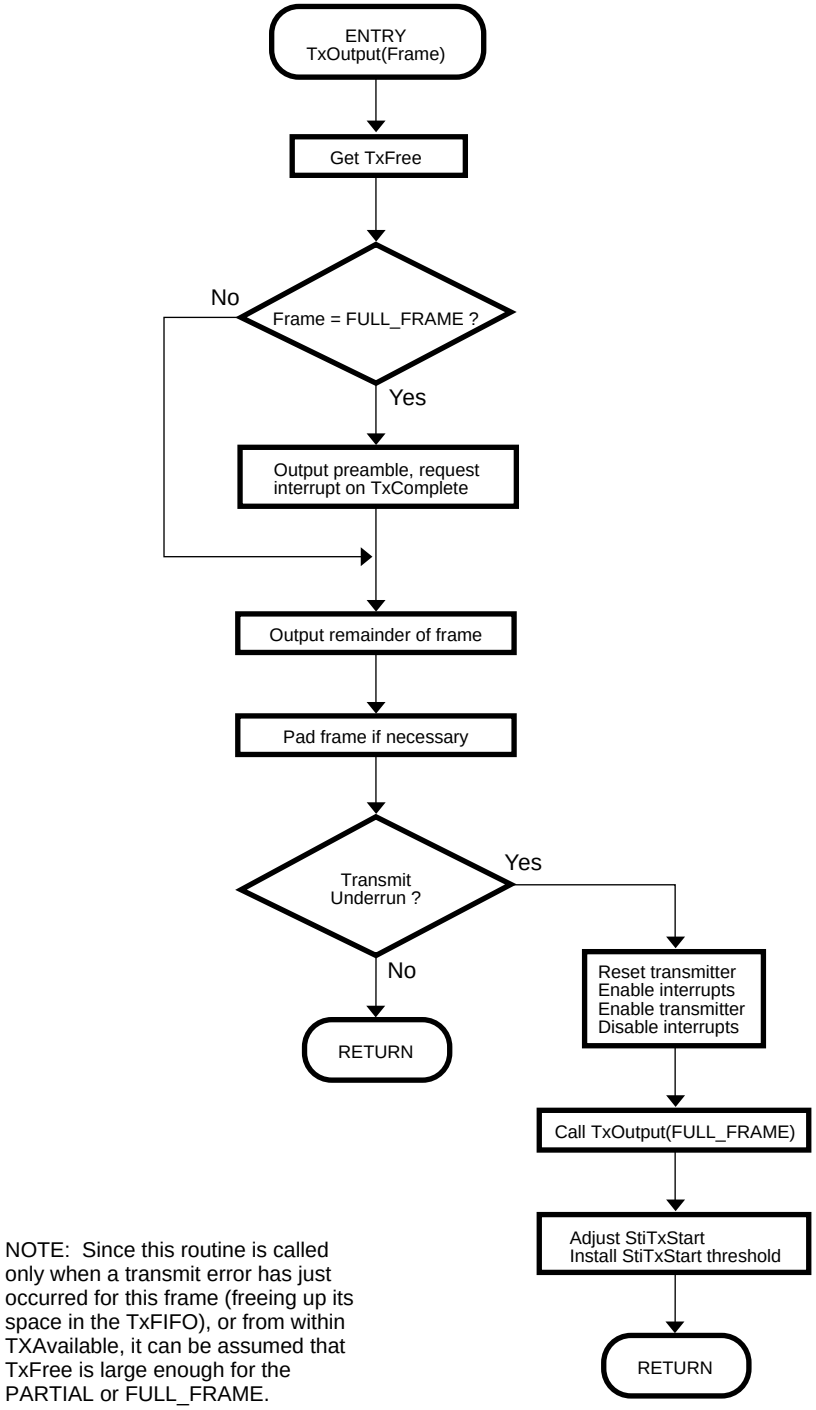


Figure A-11. Transmit Output

Interrupt Service Routines

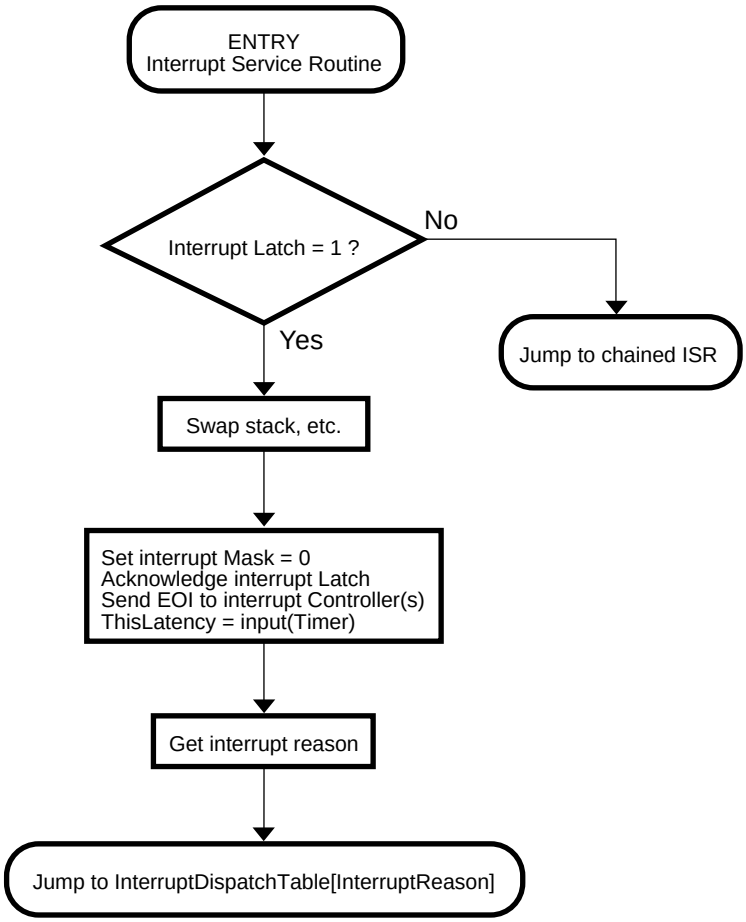


Figure A-12. Interrupt Service Routines (1 of 7)

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

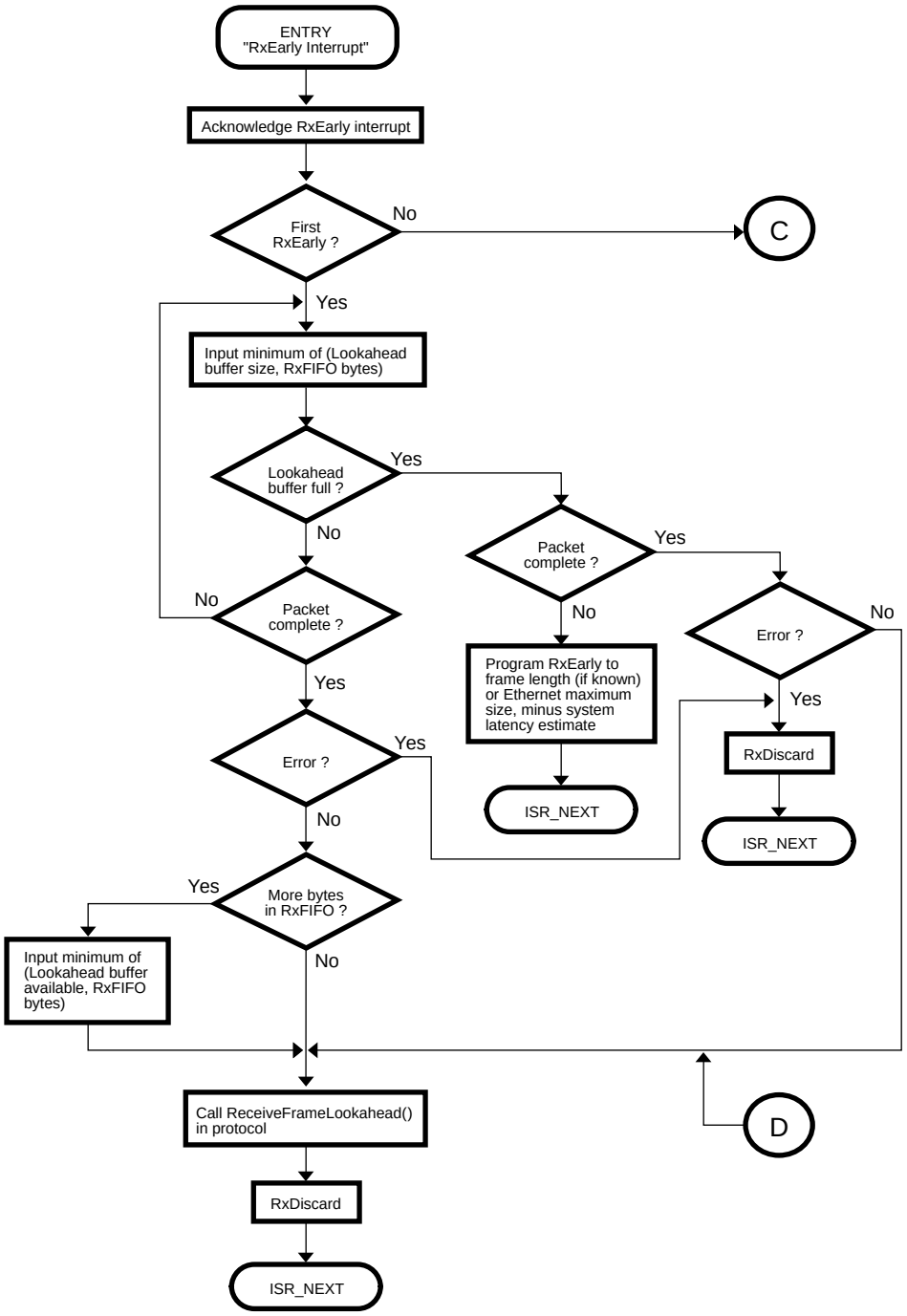


Figure A-13. Interrupt Service Routines (2 of 7)

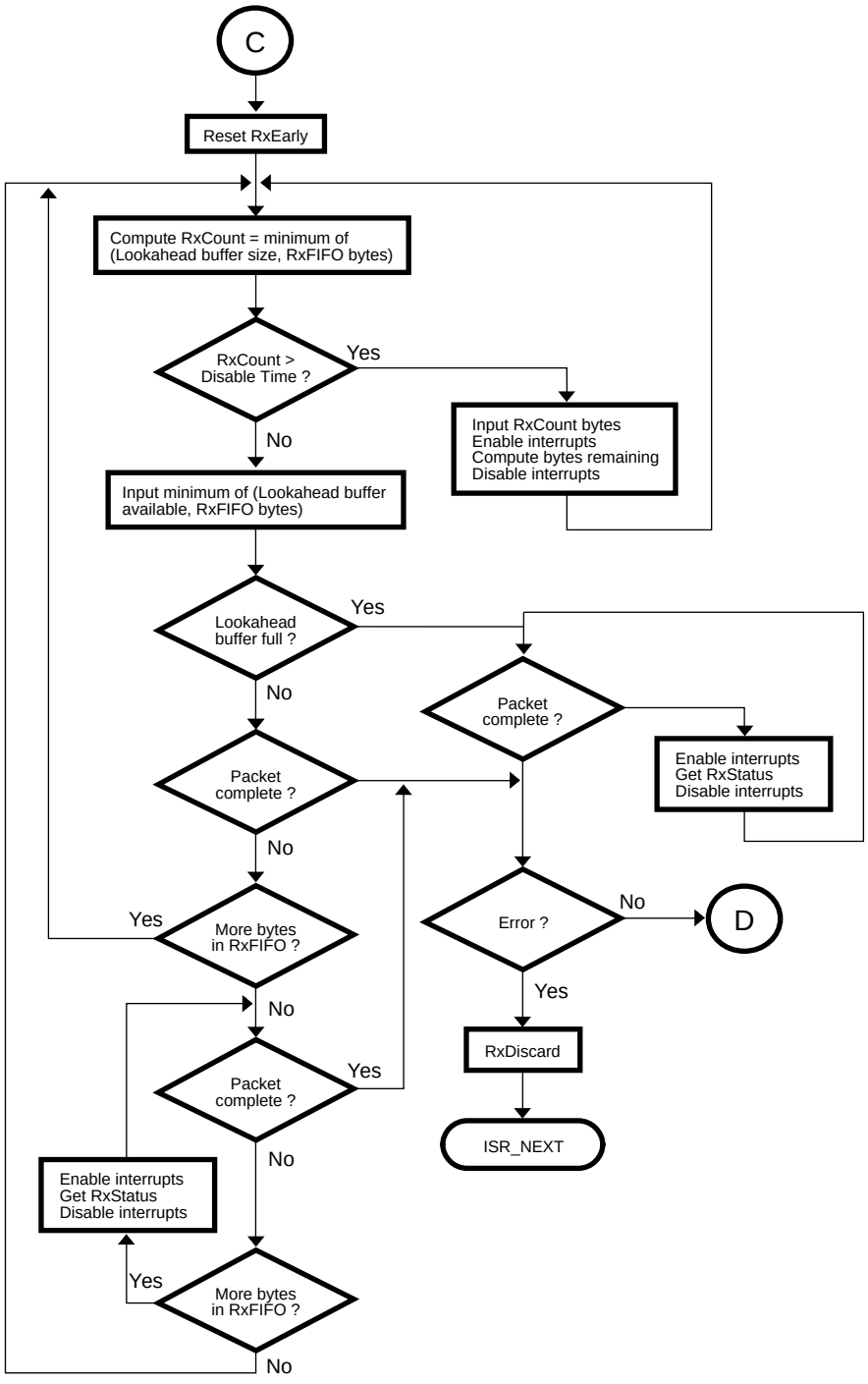


Figure A-14. Interrupt Service Routines (3 of 7)

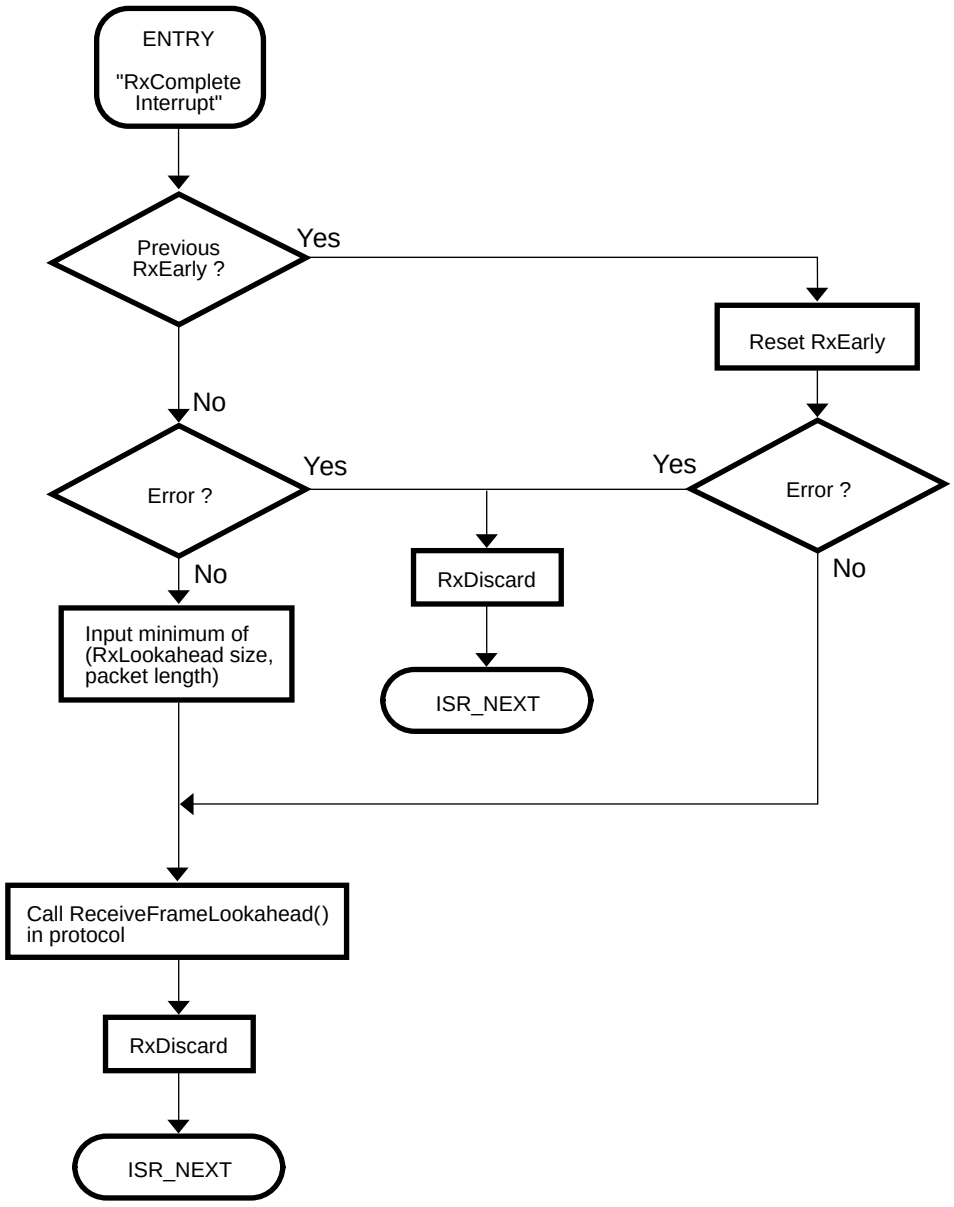


Figure A-15. Interrupt Service Routines (4 of 7)

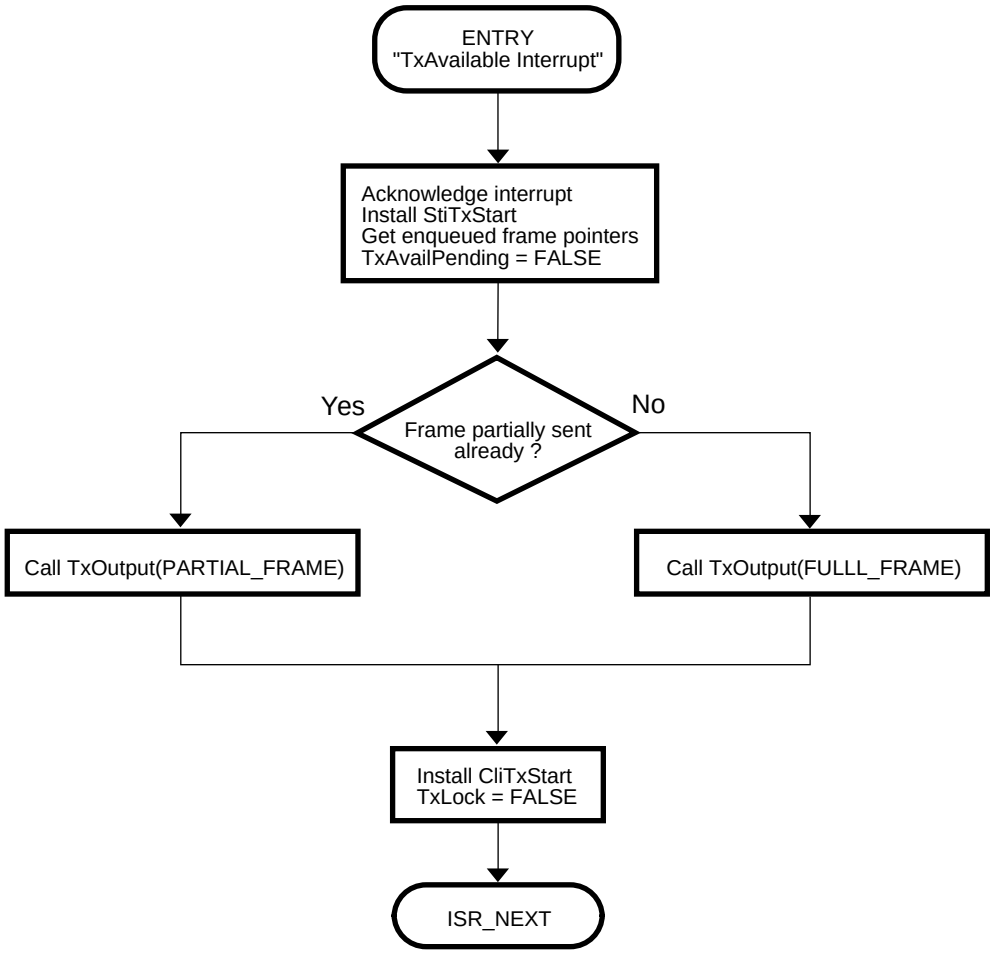


Figure A-16. Interrupt Service Routines (5 of 7)

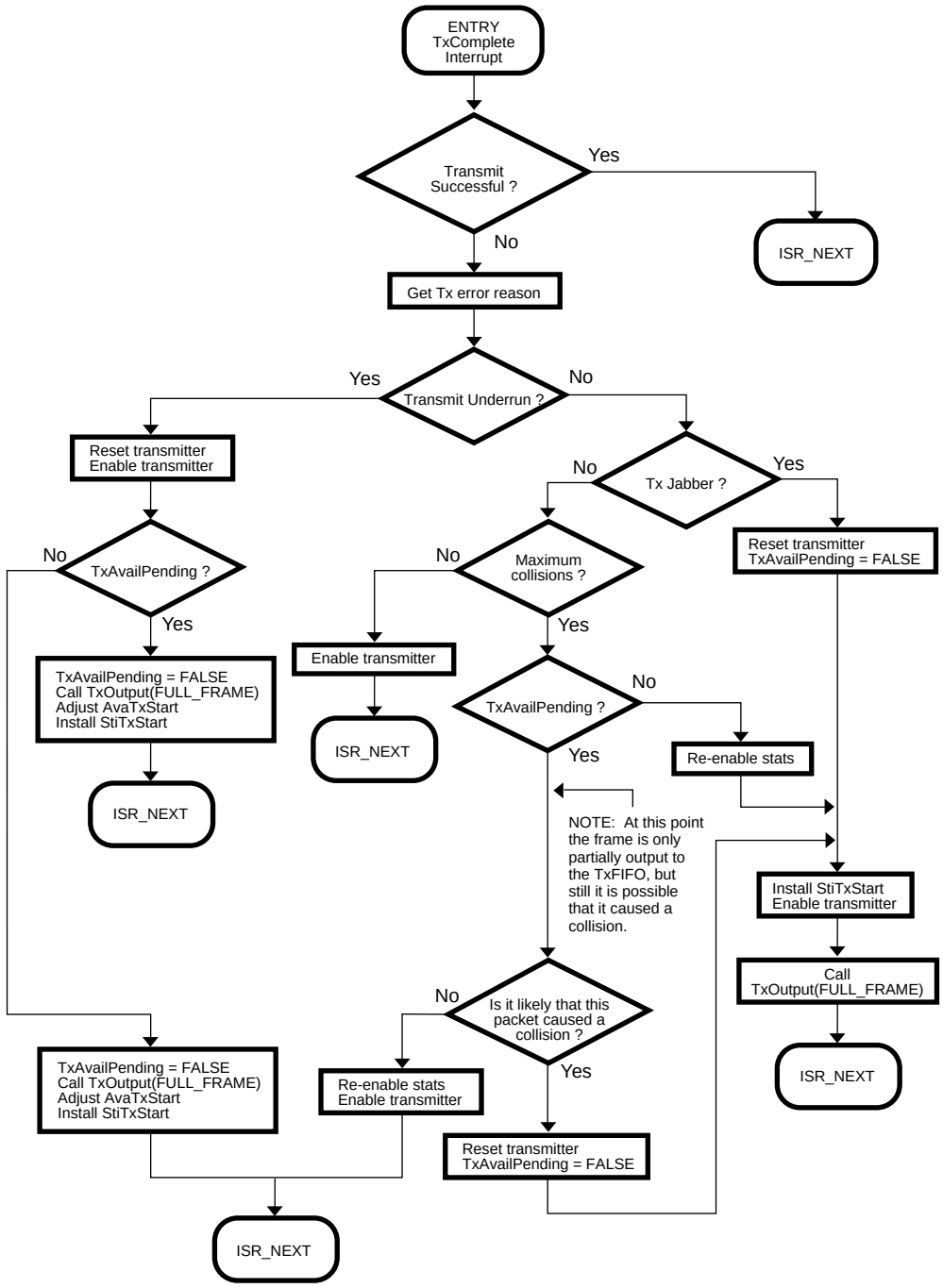


Figure A-17. Interrupt Service Routines (6 of 7)

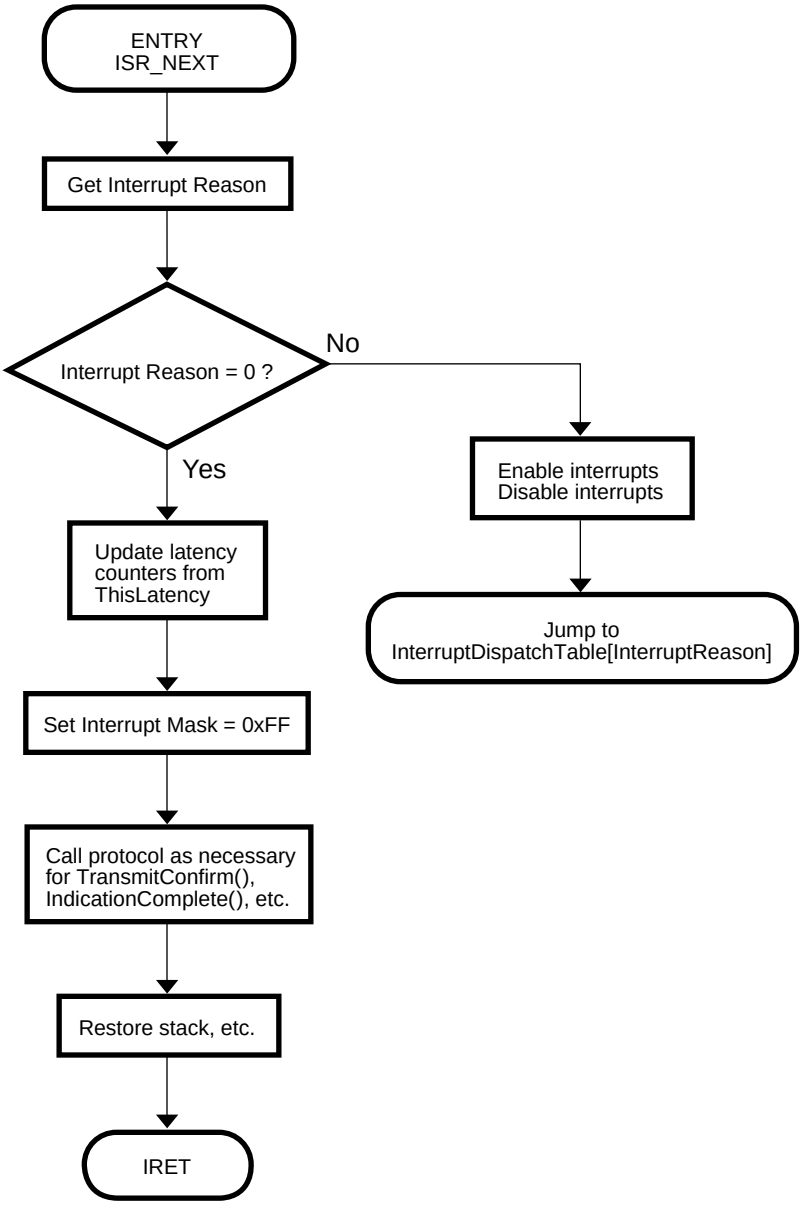


Figure A-18. Interrupt Service Routines (7 of 7)

Timer Interrupt

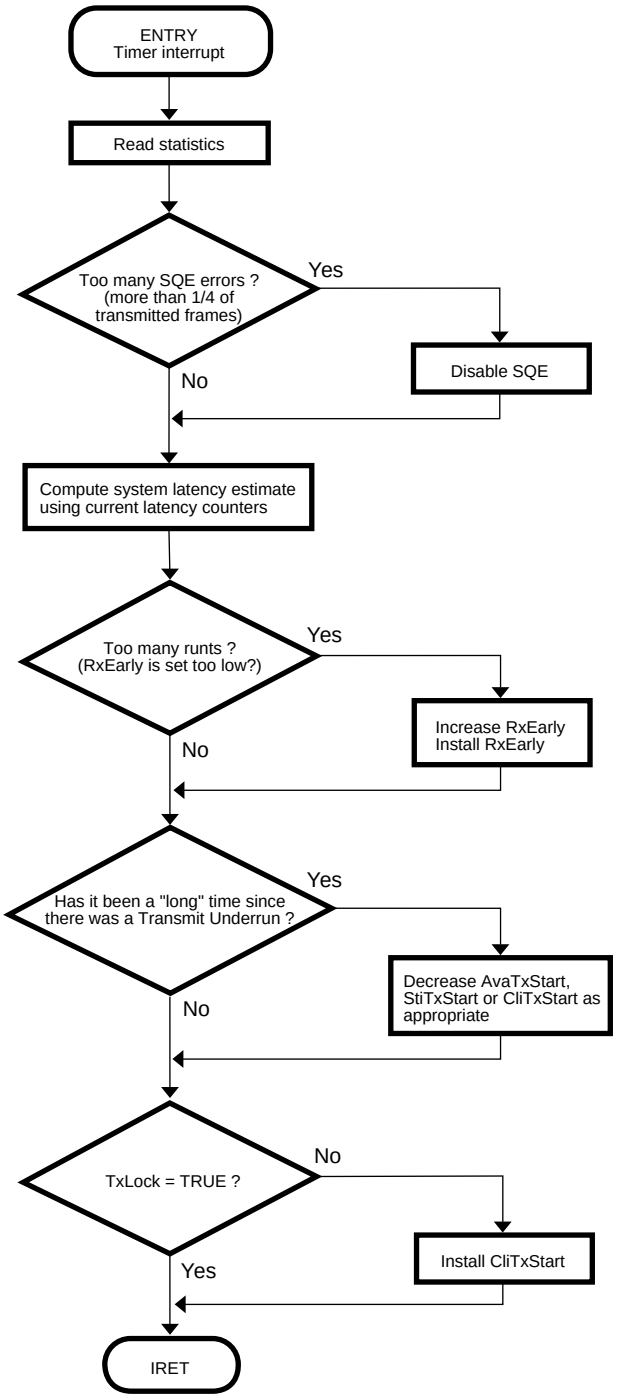


Figure A-19. Timer Interrupt