

# 3Com

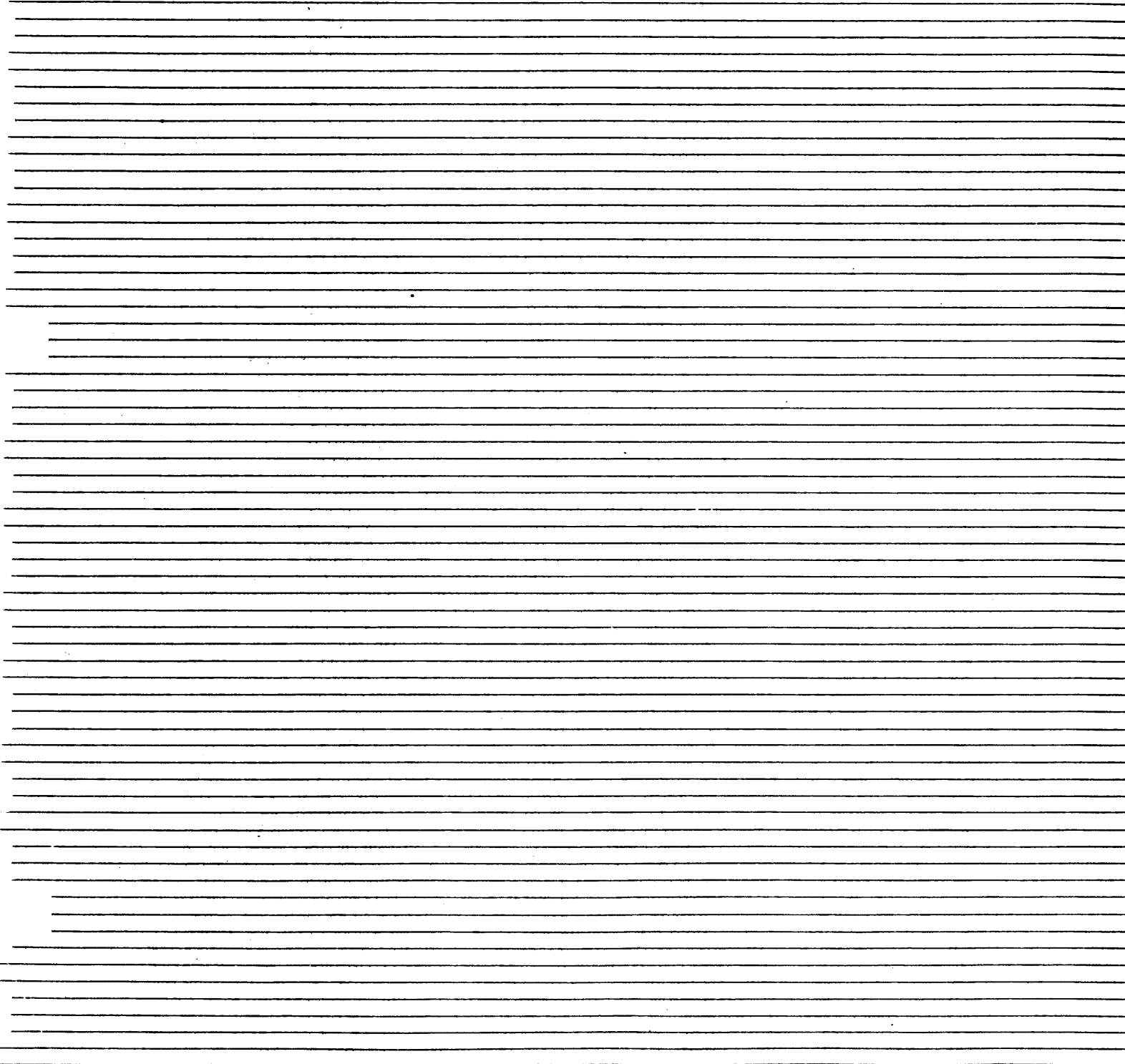
computer communication compatibility

---

## 3C400 Multibus

## Ethernet Controller

## Reference Manual



This is an addendum to the  
Model 3C400  
MULTIBUS Ethernet (ME) Controller  
Reference Manual  
of  
May 18, 1982

CORRECTIONS:

Page 4-6: The "address match" status bit is inverted. In other words, the bit is 0 if the destination address of the received packet is equal to the station address, and 1 if it is not.

Page 4-6: The "broadcast" status bit is also inverted. In other words, the bit is 0 if the the destination address is the broadcast address (all ones), and 1 if it is not.

CLARIFICATIONS:

Page 4-5: The JAM bit is cleared by setting it (writing a one into the JAM bit).

Page 4-5: The TBSW bit remains 1 during all JAM processing. It does not become 0 again until the packet has been successfully transmitted.

Page 4-6: There is no status bit indicating a multicast packet which is not a broadcast packet. Such a packet is identified by a broadcast status bit equal to 1 (false) and the multicast bit in the destination address being 1 (true). Multicast packets will only appear in receive buffers if the receiver is enabled to accept them.

Page 4-7: Broadcast is a special case of multicast. Enabling the receiver for multicast (modes 3,4,5) includes the broadcast address.

Page 4-7: The multicast bit of the destination address is the least significant (low order) bit of the first byte of the packet. In a receive buffer, this is the byte immediately following the buffer header word.

Page 4-12: The number of retransmissions normally exceeds 15 only when the Ethernet is broken. When this occurs the only way to get the ME to return the transmit buffer to the processor is to reset the controller by setting bit 8 of MECSR.

**Model 3C400**  
**MULTIBUS Ethernet (ME) Controller**  
**Reference Manual**  
**May 18, 1982**

**ABSTRACT**

This document describes the 3Com 3C400 Multibus Ethernet Controller that connects any Multibus compatible system processor to a DEC-Intel-Xerox Ethernet Communication System.

**3Com Corporation 1390 Shorebird Way, Mountain View, California 94043 USA**  
**(415) 961-9602 Telex: 345546**

**REQUEST FOR READERS COMMENTS**

We want you to have documentation that meet your needs. Please use this form to participate directly in the documentation process and help us improve our publications.

Tell us of your experience regarding the usability, accuracy, readability, organization, and completeness of **this publication only**. If you have comments on the product this publication describes or if you wish to order publications, please contact your 3Com Representative directly.

Please fill out the form below and mail to 3Com. Thank you.

NAME \_\_\_\_\_ DATE \_\_\_\_\_

TITLE \_\_\_\_\_

COMPANY NAME/DEPARTMENT \_\_\_\_\_

ADDRESS \_\_\_\_\_

CITY \_\_\_\_\_ STATE \_\_\_\_\_ ZIP CODE \_\_\_\_\_

1. Please describe any errors you found in this document (include page number).

2. Does this document cover the information you expected or required? Please make suggestions for improvement.

3. Is this the correct level documentation for your needs? Any suggestions for other types of publications needed?

4. Did you have any difficulty understanding the material? Where?

**TABLE OF CONTENTS**

**CHAPTER 1 - 3COM ME CONTROLLER SPECIFICATIONS**

- 1.1 DESCRIPTION
- 1.2 ME FEATURES
- 1.3 ME SPECIFICATIONS

**CHAPTER 2 - BACKGROUND INFORMATION**

- 2.1 BASIC ETHERNET SUBSYSTEMS
- 2.2 EXAMPLE FILE TRANSFER
- 2.3 HOW 3COM PRODUCTS IMPLEMENT ETHERNET  
FOR MULTIBUS COMPATIBLE SYSTEMS
- 2.4 ETHERNET OPERATION
- 2.5 TRANSMISSION SUBSYSTEM
- 2.6 ME CONTROLLER SUBSYSTEM

**CHAPTER 3 - PHYSICAL DESCRIPTION**

- 3.1 ENVIRONMENT
- 3.2 PACKAGING
- 3.3 BLOCK DIAGRAM

**CHAPTER 4 - 3COM ME PROGRAMMING**

- 4.1 MEMORY ALLOCATION
- 4.2 TRANSMIT/RECEIVE
- 4.3 PROGRAMMING ANOMALIES
- 4.4 OPERATION

**CHAPTER 5 - INSTALLATION AND CONFIGURATION CONSIDERATIONS**

- 5.1 INSTALLATION CHECKLIST
- 5.2 ETHERNET ADDRESSING CONSIDERATIONS

**CHAPTER 6 - FACTORY WARRANTY**

**APPENDICES**

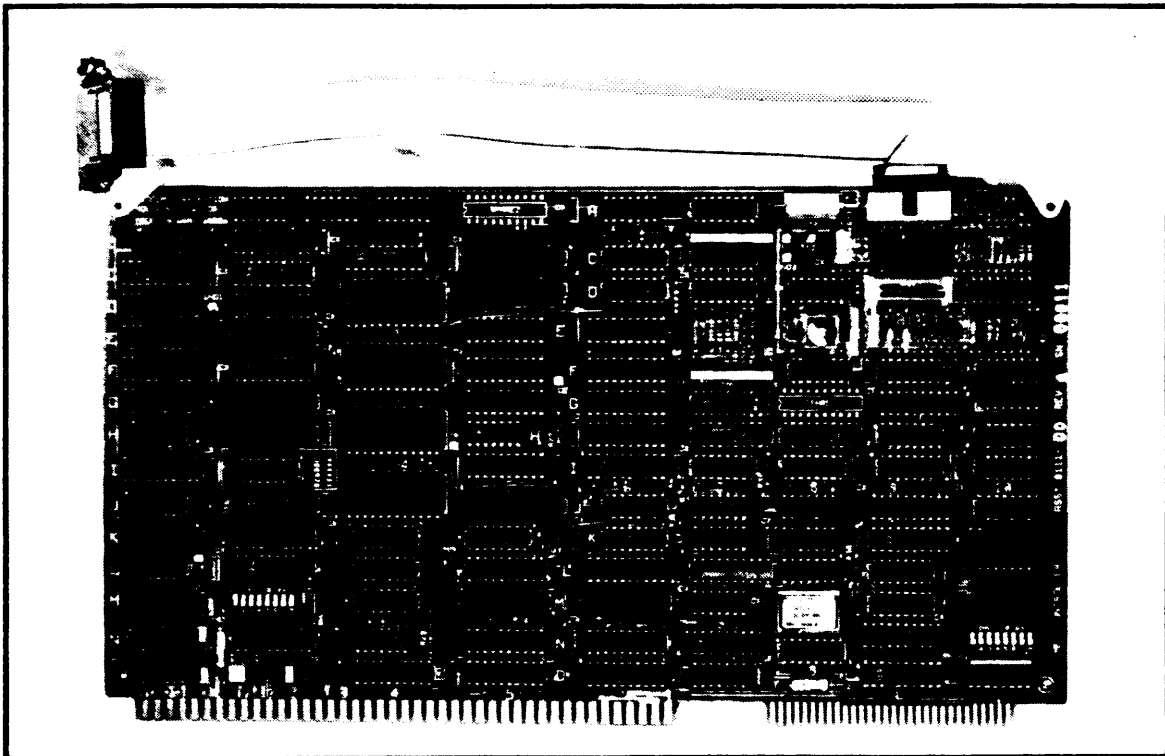
- A. ORDERING INFORMATION
- B. BIBLIOGRAPHY

**CHAPTER 1**

**3COM ME CONTROLLER SPECIFICATIONS**

**1.1 DESCRIPTION**

The 3C400 Multibus Ethernet (ME) Controller provides the connection to Ethernet for any Multibus compatible system processor. It consists of one Multibus (IEEE-796) board that plugs into the Multibus. (See Figure 1-1 below).



**FIGURE 1-1. 3COM MULTIBUS ETHERNET CONTROLLER BOARD SET**

Connection from the 3C400 ME Controller to Ethernet is made via the 3Com Ethernet Transceiver and Transceiver Cable, or any other Ethernet compatible transceiver and cable.

The 3C400 Controller, 3C100 Ethernet Transceiver and the 3C110 Transceiver Cable conform to the Ethernet specifications, version 1.0, published by DEC, Intel, and Xerox on 30 September, 1980. When coupled with customer supplied driver software, they implement layers one (physical) and two (data link) of the **International Standards Organization Reference Model for Open Systems Interconnection**. Any **Multibus compatible system processor so equipped will be compatible with any other Ethernet-based system at the physical and data link levels.**

## 1.2 ME FEATURES

- Compatible with 10 megabit-per-second DEC, Intel, Xerox Ethernet.
- Compatible with Multibus (IEEE-796).
- Connects to Ethernet using 3Com Ethernet Transceiver and Transceiver Cable or any other transceiver and cable that conform to the Ethernet specification.
- Controller, transceiver, and cable together provide a complete hardware implementation of the Ethernet specification except for multicast address comparison and random number generation for retransmission timing.
- Includes 8K byte dual-ported memory which appears in Multibus memory space. Transmission between the dual-ported memory and Ethernet do not consume Multibus cycles, allowing concurrent processing.
- Three 2K byte buffers can each handle maximum packet size allowed by Ethernet specification. One buffer is dedicated to transmission, two to reception of Ethernet packets.
- Under software control, each packet buffer may be independently connected to either the Ethernet or the Multibus.
- Multibus-addressed buffers allow in-place packet assembly, processing and multiplexing.
- Can receive minimally spaced packets.
- Controller can be selectively enabled to recognize packets containing station address, broadcast packets, multicast packets or all packets.
- Ethernet address assigned by 3Com is held in PROM on controller and can be referenced or replaced by software when loading address recognizer.
- Manchester decoding using phase-locked loop circuitry.
- 32-bit CRC generated on transmission and verified on reception.
- Hardware generation and removal of preamble.
- Hardware retransmission timing with random number supplied by software.
- Hardware detection of oversized and undersized packets and alignment errors.
- Functions as 16-bit memory slave and is compatible with 8-bit and 16-bit masters.
- Contained on one Multibus board.



### 1.3 ME SPECIFICATIONS

#### Compatibility

Ethernet	Conforms fully to Ethernet specification, version 1.0, published 30 September, 1980, by DEC, Intel, and Xerox.
Multibus	Conforms fully to Multibus specification. Functions as 16-bit memory slave. Compatible with 8-bit and 16-bit masters.
IEEE-796	Compliance is D16 M24 V0 (16-bit transfers, 24-bit addressing, non-bus vectored interrupts).

#### Functions

Serial/parallel and parallel/serial conversions.  
Transmit and receive buffering.  
Framing of packets.  
Manchester encoding and decoding.  
Address recognition  
Collision and error detection.  
Preamble generation and removal.  
Carrier sense and deference.  
Backoff and retransmission timing.  
Collision fragment filtering.  
Frame check generation and detection.  
Alignment error and overrun filtering.

#### Memory

Transmit	One 2K byte dual-ported RAM memory.
Receive	Two 2K byte dual-ported RAM memories.
Control & Status	Registers which occupy 2K bytes of Multibus address space.
Byte Ordering	Low order first or high order first, switch selectable.
Address	Occupies 8K bytes of Multibus memory address space. Starting address set by switches at any 8K byte address boundary in range 0 to 1016K bytes.

**Timing**

Bit Rate	10 million bits per second
Packet Spacing	9.6 microseconds minimum
Transmit Delay	1 microsecond typical, without deference
Receive Delay	3 microseconds typical
Jam Time	Transmits 32 bits of zeroes when collision is detected.

**Ethernet Packet Format**

Length	512 bits minimum, 12,144 bits maximum, excluding 64-bit preamble.
Format	Destination Address.....48 bits Source Address.....48 bits Type.....16 bits Data.....8n bits where n=46 minimum, 1500 maximum Frame Check Sequence.....32 bits
Frame check sequence	32 bit CRC generated on transmit, verified on receipt.
Preamble	Generated and removed by controller.

**Address Recognition****Ethernet Address**

Unique 48-bit address assigned by 3Com. Held in socketed PROM on controller, appears in Multibus address space.

**Address Recognizer**

Station Ethernet address held in RAM memory, may be loaded from Ethernet address PROM or with an address supplied by software. Can be selectively enabled by software to recognize packets containing:

- Station or broadcast address
- Station, broadcast, or multicast address
- Any address

**Error Handling**

Controller can be selectively enabled by software to reject packets with FCS, alignment or range errors.

**Interrupts****Interrupt Conditions**

Selectively enabled by software:

- Transmit done
- Receive buffer A full
- Receive buffer B full
- Collision (jam)

**Priority levels**

All interrupts use a common priority level, jumper selectable from INTO to INT7.

**Software Functions**

The following functions must be performed by customer software:

- Loading of Ethernet address
- Multicast address comparison
- Random number generation for retransmission timing after collision.

**Installation****Size**

One Multibus-standard board

- 30.5cm X 17.1cm
- 12in X 6.75in

Slots	Requires one slot.
Power	5A at +5V 0.5A at +12V for transceiver
Bus loading	One DC load
Transceiver cable connector	Ethernet-standard female 15-pin "D" subminiature connector attached to the controller via cable.
Transceiver cable	Uses Ethernet-standard transceiver cable which must be ordered separately.
Ethernet address	Unique address supplied by 3Com for each controller, contained in onboard PROM.
<b>Operating Environment</b>	
Temperature	5° to 55° C
Humidity	10% to 90% without condensation

## CHAPTER 2

### BACKGROUND INFORMATION

This chapter covers some background information about both the Ethernet and the 3Com ME Controller.

In the last decade, computers have grown from a luxury to a necessity in most businesses. Similarly, in the next decade, **inter-computer communication** will grow from a luxury to a necessity.

The "Ethernet" network, developed for machine-machine communication, was pioneered at Xerox Corporation as an appropriate implementation for inter-computer communications. In use since 1974, the Ethernet has evolved to an industry standard, documented in the Ethernet Specification, published September 30, 1980 by DEC, Intel, and Xerox.

The benefits of modern computerized workstations are now magnified as they communicate information to other devices at 10 million bits-per-second over the Ethernet. What's more the Ethernet network can be tailored to end user's needs and workstations once the Ethernet coaxial cable is in place. This means multiple workstations can share

resources such as:

Word processors	Data bases
Printers	Process control stations
Electronic mail systems	Array processors
Graphics stations	Laser printers
Transaction workstations	Etc.

Individual workstations and shared resources are plugged into Ethernet Information Outlets in the wall the same way telephones are plugged into telephone wall-outlets. However, the Ethernet's 15-pin connector is more complex than a telephone connector.

Each Ethernet device is assigned a unique address (like a unique telephone number), therefore, it can be moved around and plugged into any convenient Ethernet information outlet. Further, all devices plugged into the Ethernet can talk to each other, by mutual agreement, similar to two people talking on the telephone.

Ethernet, due to its **standardized** physical and logical protocol, allows users to mix and match equipment from **multiple vendors**.

In the future a voice capability will probably be integrated into Ethernet for store-and-forward voice communications to complement electronic mail.

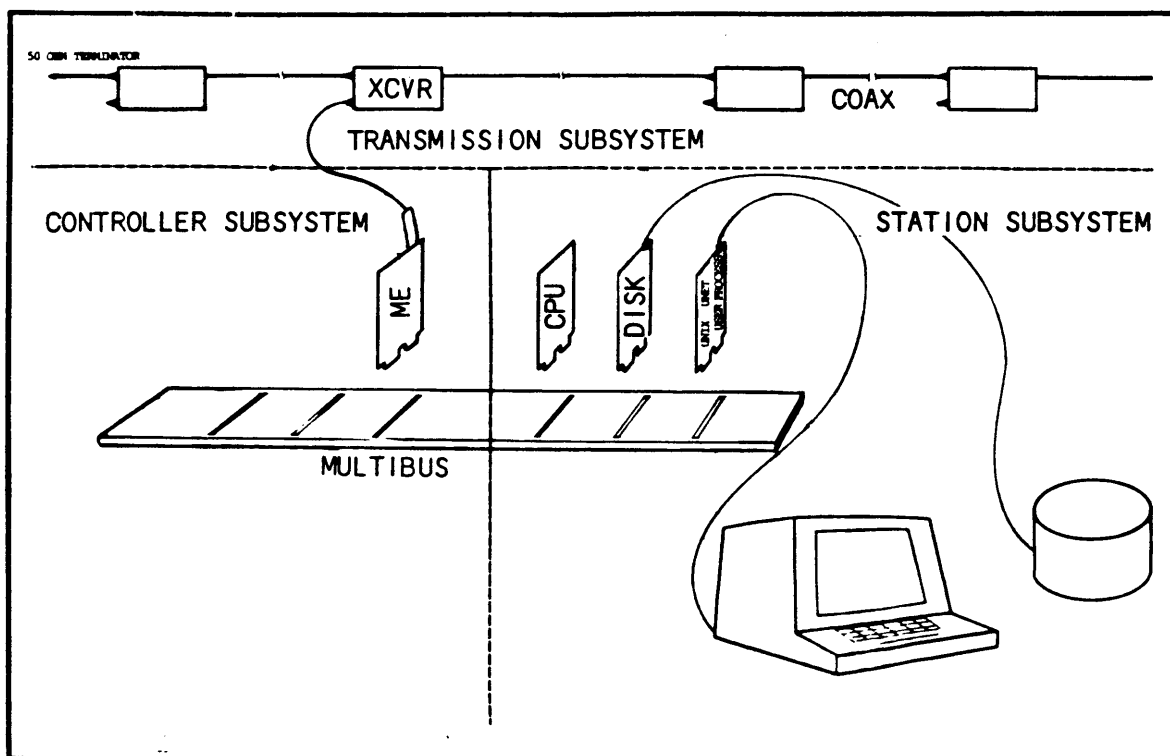
## 2.1 BASIC ETHERNET SUBSYSTEMS

The Ethernet is a bus-oriented communication system that supports up to 100 stations using a 50 ohm coaxial cable as the bus.

Figure 2-1 below shows the basic parts of a typical Ethernet system, with workstations connected to the Ethernet coaxial cable.

The **Transmission Subsystem** is made up of 50 ohm coaxial cable, terminators, transceivers, and transceiver cables.

The **Controller Subsystem** is the set of controller boards and the software



**FIGURE 2-1. BASIC ETHERNET SUBSYSTEMS**

supporting them.

The **Station Subsystem** is everything else associated with the station, such as, the terminal, processor, disk, and higher level protocol software.

These three subsystems are discussed again later in this chapter, in terms of 3Com product implementation.

Meanwhile, the following example describes how a file of information is transferred from one device to another using Ethernet.

## 2.2 EXAMPLE FILE TRANSFER

(This is an example text file transfer using a File Transfer Program running on the host processor.)

1. The terminal user runs the File Transfer Program, connects to the receiver, and specifies the file to be transferred.
2. The file's characters are mapped into device-independent virtual characters (by software) to meet protocol specifications.
3. The mapped character stream is then routed to a virtual circuit set up between the two devices.



4. The virtual circuit protocol software breaks the character stream into packets for transmission. (It also retransmits corrupted packets, and limits data rate to avoid overruns.)
5. The packets are then passed to the Ethernet driver software.
6. The Ethernet driver then copies the packet into a packet buffer and tells the controller to transmit it.
7. The controller waits until the coaxial cable is not in use, then transmits the packet.
8. The Ethernet transceiver receives the packet's bit stream and injects it onto the coaxial cable. (If the transceiver detects a collision, it signals the controller to retransmit.)
9. The receiving station recognizes its address and reverses the above procedure: bits are received by the transceiver, fed to the controller, passed to software that reassembles the packets, maps the characters, and stores the data.

### 2.3 HOW 3COM PRODUCTS IMPLEMENT ETHERNET FOR MULTIBUS COMPATIBLE DEVICES

For a complete local computer network, there are only three additional components needed by Multibus compatible systems:

1. **3Com Ethernet Transceiver** - fully conforms to published Ethernet specifications and connects directly to the Ethernet coaxial cable.
  
2. **3Com Ethernet Controller** - plugs directly into the Multibus
  
3. **Higher-level Protocol Software** - providing high-level network protocol services - including data link drivers. 3Com's UNET Software is UNIX compatible and provides the Internet Protocol (IP), Transmission Control Protocol (TCP), file transfer protocol (UFTP), electronic mail protocol (UMTP), virtual terminal protocol (UVTP), etc.

## 2.4 ETHERNET OPERATION

The Ethernet is a carrier sense, multiple access transmission system with collision detection (CSMA/CD). To transmit a packet, a station waits for quiet on the network (defers). When the network is quiet, it starts to transmit the packet.

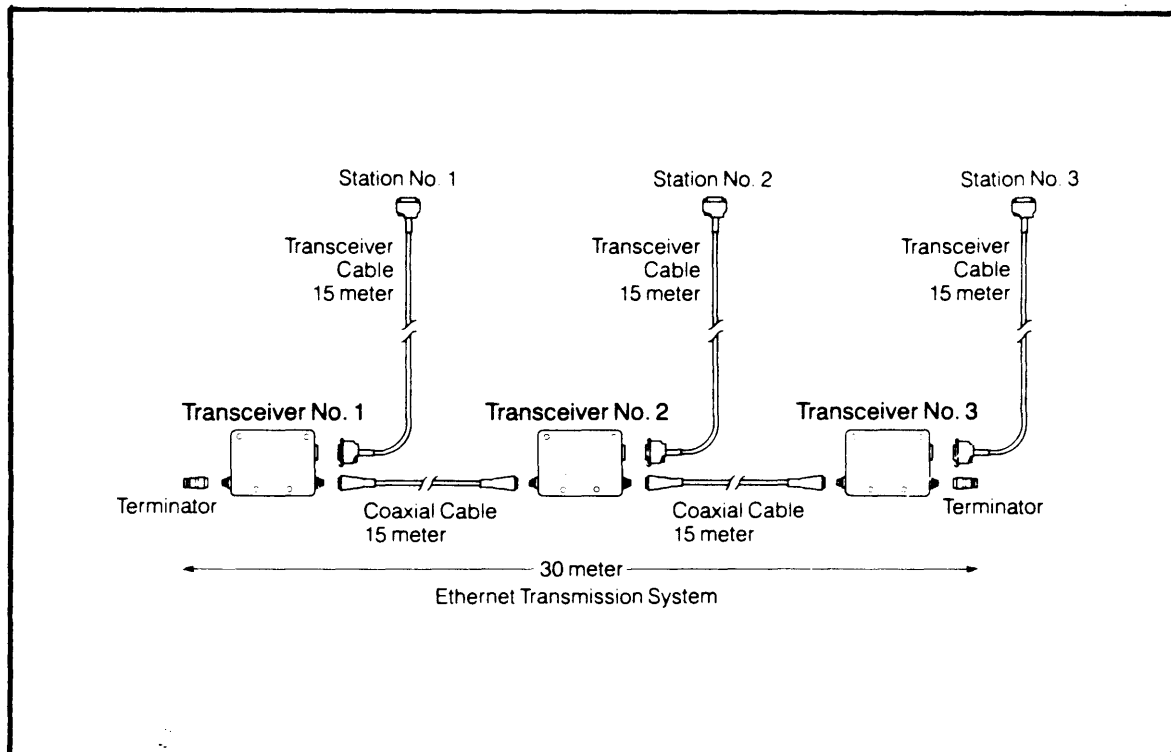
During packet transmission, the station also watches for collisions with other transmitters; these may occur within one round-trip time through the network. The station is said to have "acquired the network" if no collision occurs in that time interval. If a collision does occur, the station transmits 4 to 6 additional bytes of data (jam) and then aborts the packet. The extra bytes insure that any other participant in the collision is sure to see it. The station then waits a random amount of time (backoff) before retransmitting (after deferring to packets in progress on the network).

## 2.5 TRANSMISSION SUBSYSTEM

The transmission subsystem, in the form of a 3Com "starter package",

(Model 3C440) is shown in Figure 2-2 below. It consists of four types of components: transceiver cables, transceivers, coaxial cable, and terminators. These are described below.

**Transceiver Cable** - The transceiver cable is a 15 meter shielded twisted pair cable that connects the controller to the transceiver. It has 4 pairs, one each for transmit, receive, collision detect, and power. It has a male 15 pin connector with lock posts on the controller end and a female 15 pin D connector with slide lock assembly on the transceiver end. Thus the cables can be concatenated to make a longer cable, up to the maximum length of 50 meters.

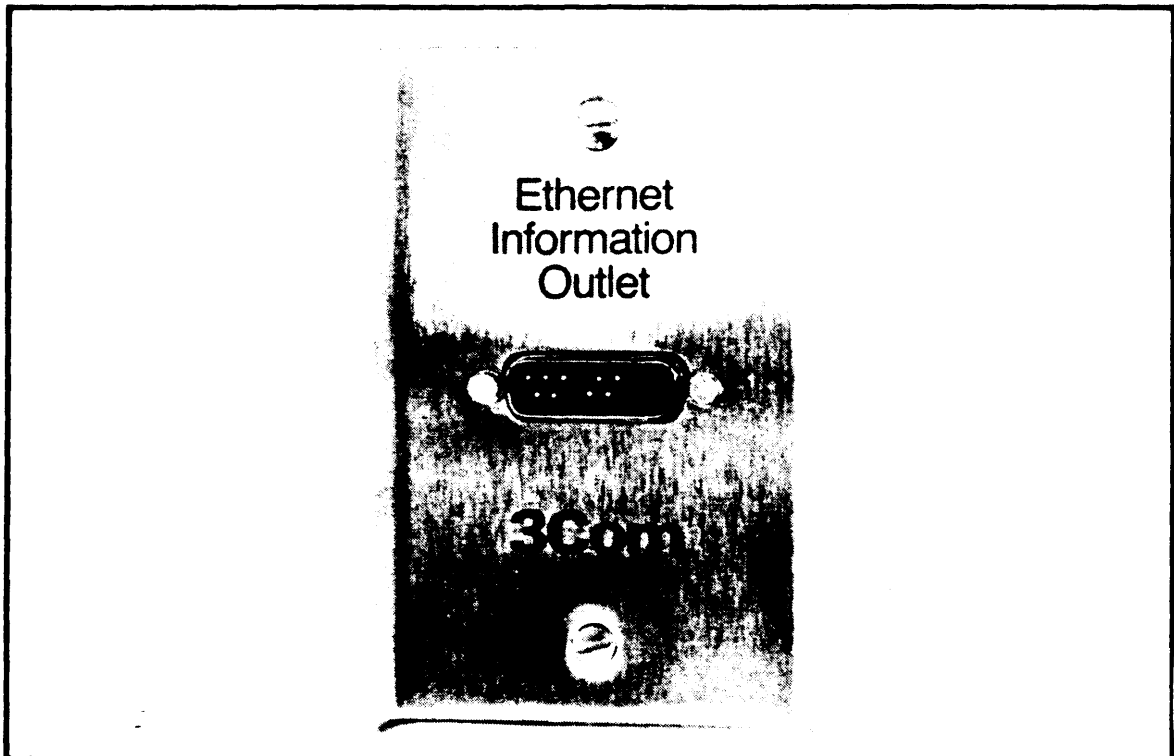


**FIGURE 2-2. 3COM ETHERNET TRANSCEIVER STARTER PACKAGE**

To minimize EMI (electro magnetic interference), the connectors have internal shields connecting the cable shield to the shell of the connector.

The male cable connector can either be brought out of the wall to the station or mounted on a cover plate providing a bulkhead disconnect at the wall. When mounted on the cover plate, it has been referred to as the "Information Outlet." (see Figure 2-3 below)

**Transceiver** - The 3Com transceiver is compatible with the DEC, Intel and Xerox Ethernet specification. It makes a high impedance connection to the common coaxial cable and provides electrical isolation between the



**FIGURE 2 - 3. INFORMATION OUTLET**

coaxial cable and the twisted pair cable.

The transceiver injects transmit signals from the controller into the coaxial cable. The transceiver also receives signals from the coaxial cable which appear on the receive lead of the transceiver cable with balanced signalling.

The receiver also provides correction for signal distortions caused by traveling through long lengths of coaxial cable.

The collision signal appears if there is a signal present from any other station on the network. When transmitting, this indicates a collision. When **not** transmitting, it indicates the presence of other signals on the network.

**Coaxial Cable** - The coaxial cable is a 50 ohm cable with multiple shields to minimize susceptibility to strong RF fields.

**Cable Connectors** - Cable sections are terminated with standard N-series connectors. Rubber boots cover the connectors to prevent multiple connections of the coaxial shield to building grounds - a potential source of ground induced noise into the coaxial shield. Coaxial cable sections are joined by insulated barrel connectors (N-Series female-female adapters).

**Terminators** - The ends of a coaxial cable segment are terminated with 50

ohm terminators with insulated outside covers.

## 2.6 ME CONTROLLER SUBSYSTEM

The ME Ethernet Controller interfaces the transceiver to the internal bus of the Multibus system to which it is connected. It performs **serial-parallel and parallel-serial conversion, buffering, CRC generation and checking, address recognition, phase encoding and decoding** - discussed below. The I/O structure and speed of the processor determine how these functions are partitioned between hardware and software (or microcode) in the Ethernet station.

**Buffering:** Most processors have bus transfer rates that are unduly stressed by the 10Mbps Ethernet bandwidth, therefore, full packet buffers are provided to keep pace with the bit rate of network traffic.

**CRC Generation And Checking:** The cyclic redundancy code (CRC) uses the 32 bit polynomial from the U.S. Department of Defense Autodin II system. The CRC function is implemented in hardware on the ME.

**Address Recognition:** The controller watches every packet that passes to determine whether to accept the packet, based on its destination address. The ME Controller implements address recognition in hardware to minimize CPU overload.

**Phase Encoding, Decoding, and Transceiver Interface:** M a n c h e s t e r

encoding is used for data transmission on the Ethernet. It has a 50% duty cycle. The first half of a bit cell contains the complement of the bit and the second half of the bit cell contains the bit.

**Phase Encoding** is done in the controller by exclusive-ORing the clock with the data. (Decoding is also performed in the controller. Partitioning of encode-decode functions into the controller rather than the transceiver minimizes wires to the transceiver while minimizing transceiver size and power dissipation.)

**Phase Decoding** in the ME Controller is done by an analog phase-locked loop technique. This technique has the advantage of tolerating more phase jitter than alternative techniques in use, twice the tolerance of a typical one-shot decoder and four times the tolerance of a typical digital state-machine decoder.

The **Transceiver Interface** contains line drivers and receivers.



**CHAPTER 3  
PHYSICAL DESCRIPTION**

**3.1 ENVIRONMENT**

The ME Controller interfaces a Multibus compatible processor to an Ethernet transceiver.

The ME Controller plugs into the same Multibus backplane as the processor and resides in the same enclosure with it. An Ethernet transceiver cable (approximately 50 feet long) connects the ME Controller to the Ethernet transceiver. The Ethernet transceiver in turn taps directly into the Ethernet coaxial cable.

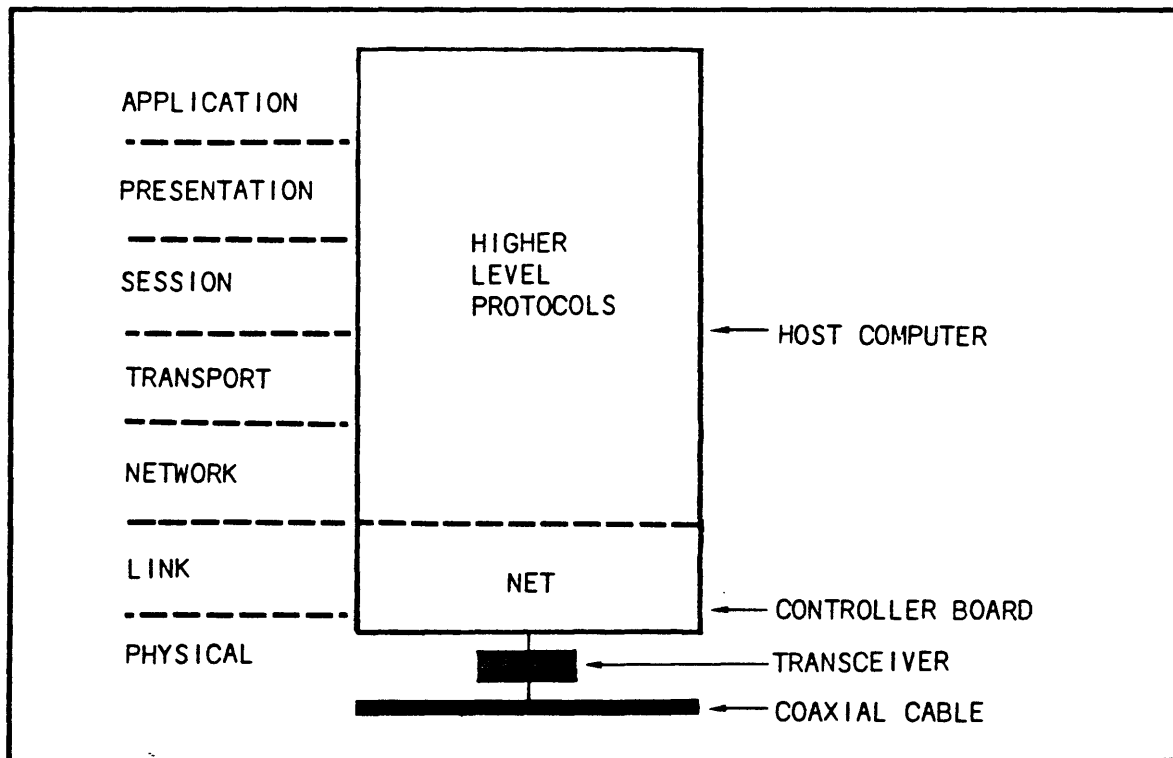
According to the International Standards Organization Open Systems Interconnection Reference Model, the ME Controller performs part of both the physical and link layer services, the first and second of seven

layers of service (see Figure 3-1 below).

The two main functions of the ME in the link layer of the ISO Model are:

**1. Data Encapsulation**

- o framing (frame boundary delimitation)
- o addressing (handling of source and destination addresses)
- o error detection (detection of physical channel transmission errors)



**FIGURE 3-1. ISO PROTOCOL HIERARCHY**

**2. Link Management**

- o channel allocation (collision avoidance)
- o contention resolution (collision handling)

In the physical layer of the ISO model, the controller performs preamble generation/removal and bit encoding/decoding (between binary and phase-encoded form).

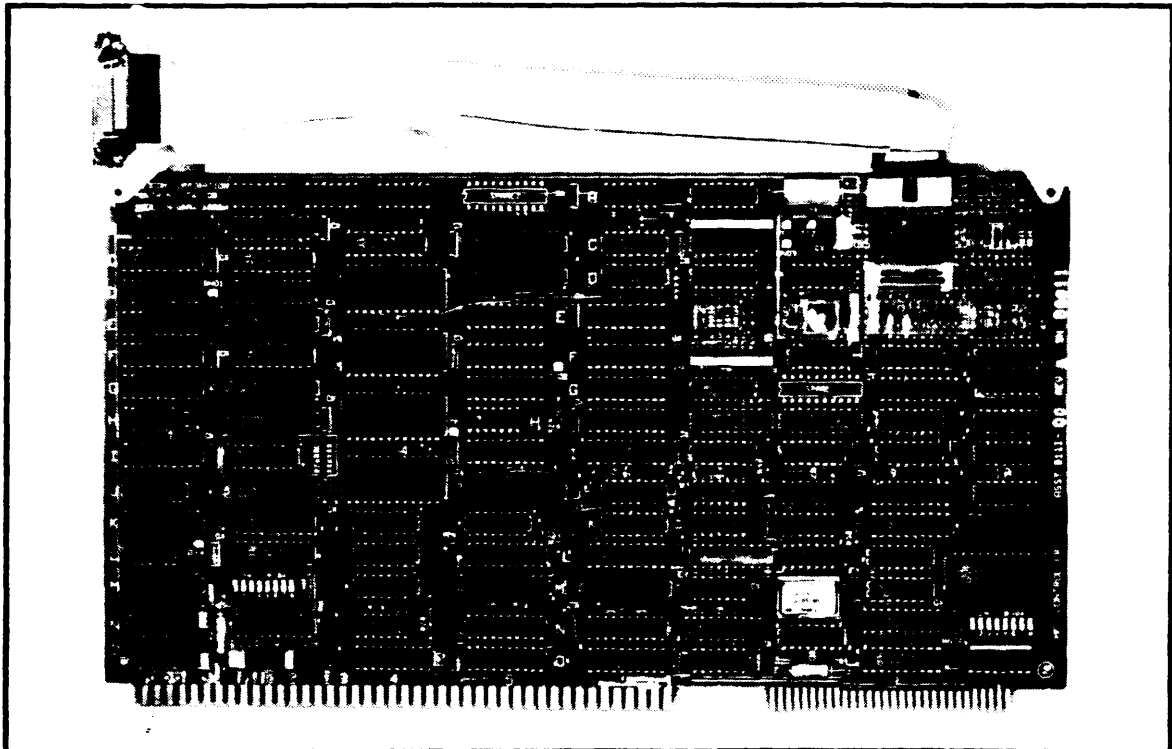
**3.2 PACKAGING**

The ME Controller is a single multibus compatible PC board whose overall dimensions are 12 inches x 6.75 inches. It uses one Multibus

backplane slot as shown in **Figure 3-2** below.

On the edge opposite the gold fingers is a 14 pin male header. A shielded cable, with four twisted pairs terminated to a female socket at one end, mates the ME Controller to the Ethernet Transceiver Cable. At this end of the shielded cable is a tin plated sub-miniature D connector. A shield drain connects to the shield of this twisted pair cable at the controller end. This shield drain is to be connected to the chassis ground of the Multibus cardcage.

The ME Controller is a high density four layer PC board with two circuit trace layers plus continuous ground and power planes.



**FIGURE 3-2. ME PACKAGING ENVIRONMENT**

The PC layout design rules followed are:

- o Ten mil traces
- o Ten mil air gaps
- o At most two traces between IC leads
- o Four layer pcb
- o Intact internal voltage and ground planes

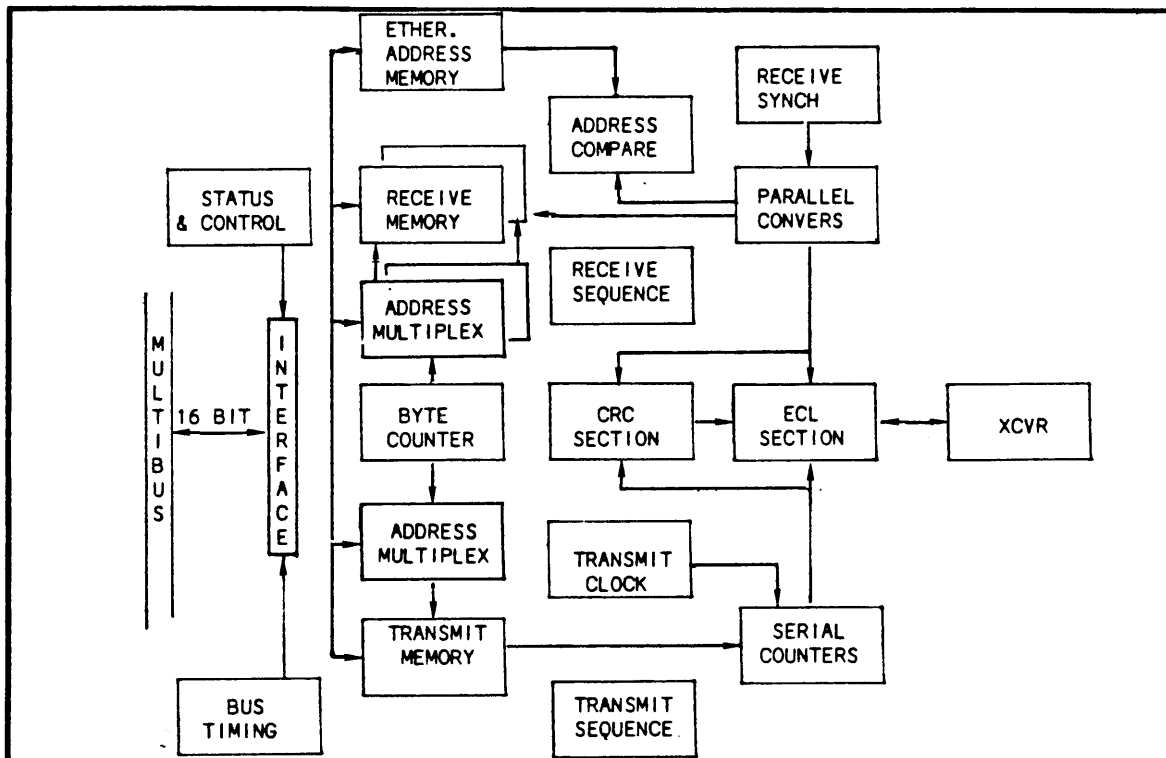
The ME Controller can be used with any transceiver that conforms to the DEC-Intel-Xerox Ethernet specifications.

### 3.3 BLOCK DIAGRAM

The major ME Controller components are shown in **Figure 3-3** below.

The ME Controller presents to the Multibus a full 16 bit interface. All bus transfer instructions to a 16 bit slave device are implemented on this controller.

The 3Com Multibus Ethernet Controller is memory-mapped and therefore appears as 8K bytes of memory on the Multibus; 4K is used for two receive buffers and 2K is used for one transmit buffer. There are four bytes of



**FIGURE 3-3. ME CONTROLLER FUNCTIONAL BLOCK DIAGRAM**

control and two words of status in the lowest 2K of memory. The actual device address is switch selectable on any 8K boundary.

On transmit, the processor loads the packet to be sent into the TRANSMIT MEMORY (see **Figure 3-3** above). Once the transmit bit is set in the control byte no further processor intervention is required. Successive bytes are taken from the TRANSMIT MEMORY and placed into the SERIALIZER where they are converted to a serial bit stream. The serialized data is fed to both the CRC (Cyclic Redundancy Check) and to the ECL SECTION which does a level conversion to be compatible with the Ethernet Transceiver. The CRC is automatically appended to the end of the transmit data packet.

On receive, the processor must first make available to the ME one or both of the RECEIVE MEMORIES available (see **Figure 3-3** above) by setting the proper control bits. Since hardware address recognition is incorporated into this design, the receive packet is further qualified. The ME can be hardware configured to receive only packets with correct physical, multicast, and/or broadcast addresses. After address recognition, the Ethernet data enters the ECL SECTION where a phase lock loop generates a proper synchronizing clock from the incoming data. Also the ECL SECTION does a level conversion to TTL signals. The serial data is sent to the PARALLEL CONVERSION logic (and converted to a byte format) and to the CRC SECTION. The CRC is calculated and compared to the CRC code appended to the end of the packet by the transmitting station. If a CRC error occurs, (Frame Check error) a flag is set in the first byte of the RECEIVE MEMORY.

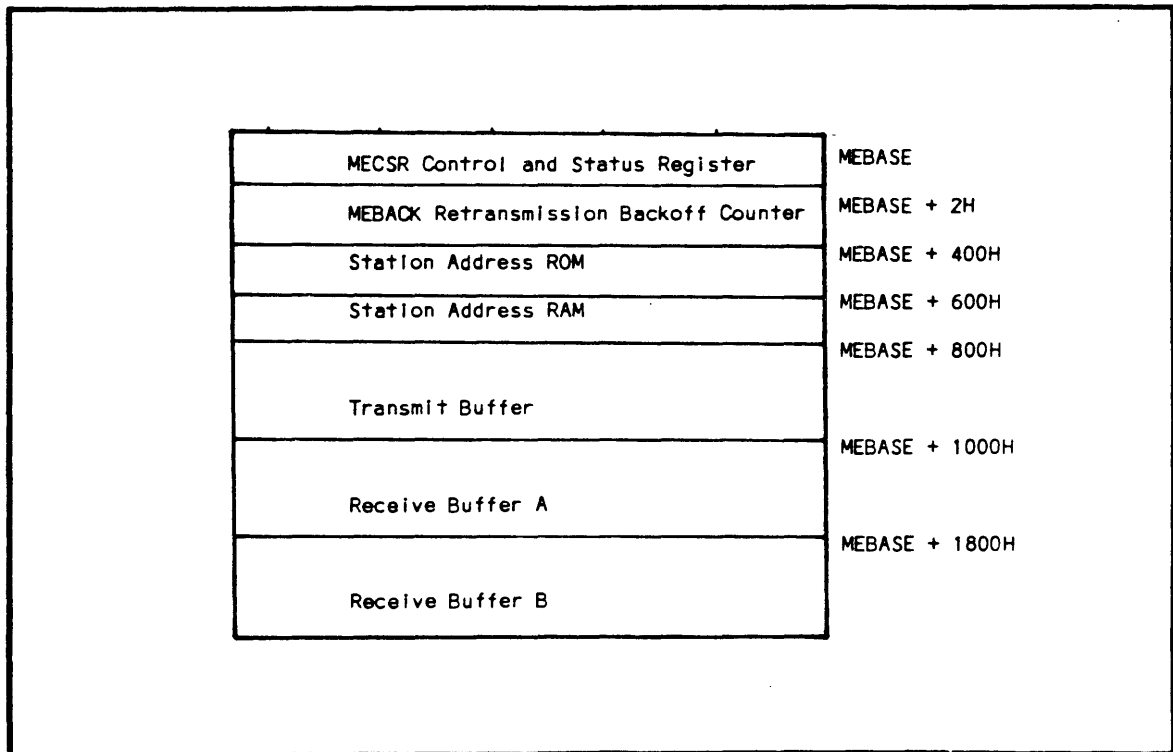
**CHAPTER 4**

**ME PROGRAMMING**

The following chapter is important for development of software drivers.

**4.1 MEMORY ALLOCATION**

The ME occupies 8K bytes of the Multibus 24 bit address space. See Figure 4-1 below. Switches on the controller select the base address of the ME memory called MEBASE. MEBASE must be aligned on an 8K byte



**FIGURE 4-1. ME BUFFER ORGANIZATION 8K BYTES, ADDRESSES IN HEX**



boundary. The controller partitions its memory into six regions as follows: (Also see **Figure 4-1** above.)

- |                                  |                  |
|----------------------------------|------------------|
| 1. Control region                | (MEBASE)         |
| 2. Station address ROM           | (MEBASE + 400H)  |
| 3. Station address RAM           | (MEBASE + 600H)  |
| 4. One 2K byte transmit buffer   | (MEBASE + 800H)  |
| 5. First 2k byte receive buffer  | (MEBASE + 1000H) |
| 6. Second 2k byte receive buffer | (MEBASE + 1800H) |

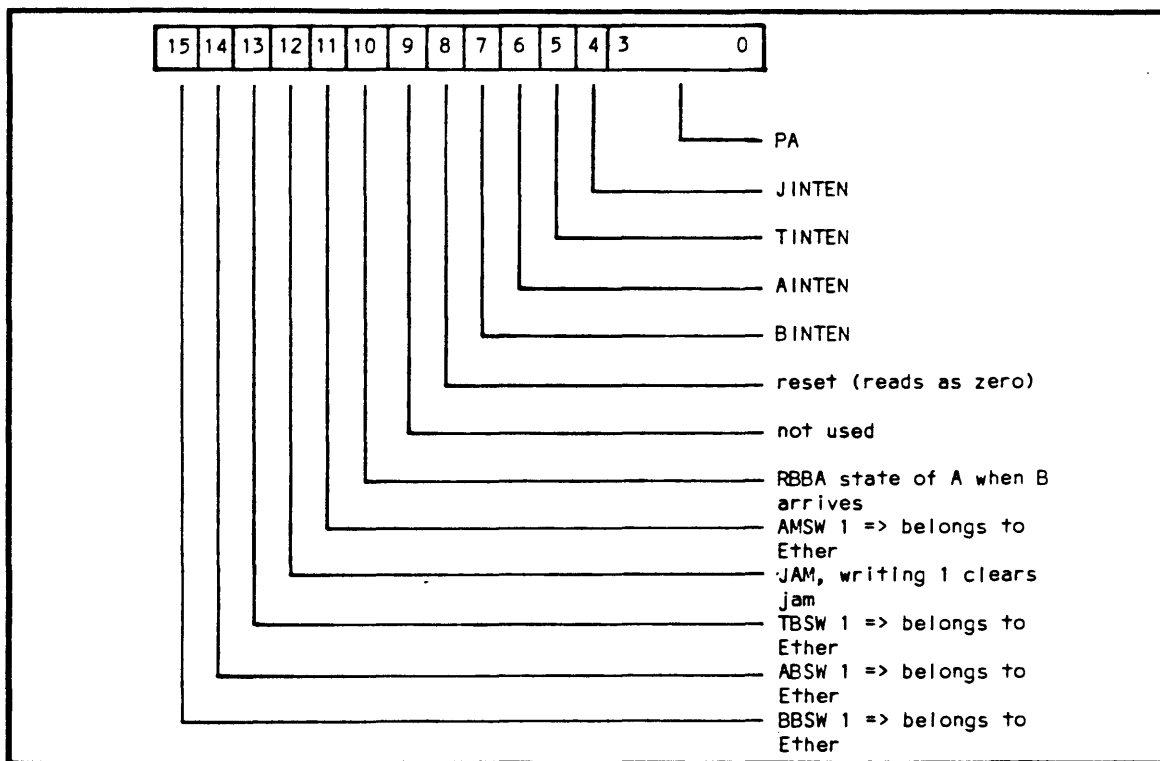
Software uses the registers in the control region to manipulate the controller and read its status. There are two registers in the control region:

1. MECSR, the control and status register, at MEBASE
2. MEBACK, the retransmission backoff counter, at MEBASE+2.

NOTE: The byte-ordering switch, TRB (See section 5.1) will affect the ordering of the bytes in both MECSR and MEBACK. If the TRB switch is OFF, MECSR and MEBACK are as shown in Figure 4-2. If the TRB switch is ON the bytes of MECSR and MEBACK are reversed.

See Figure 4-2 below.

The station address ROM, at MEBASE+400H, holds the six byte station address of the controller. The address recognition circuitry uses the station address RAM, at MEBASE+600H, to compare the destination address of packets from the ether when deciding whether to accept a packet. Software must write the station address into the RAM and "give" the address to the controller by writing one into AMSW. Any further references to the address memory are ignored until reset.



**FIGURE 4-2. MECSR=MEBASE, THE MULTIBUS ETHERNET CONTROL AND STATUS REGISTER**

## 4.2 TRANSMIT/RECEIVE

The transmit buffer starts at  $MEBASE+0x800$  (see Figure 4-1). The first word of the buffer is MEXHDR, the transmit buffer header, as shown in Figure 4-3 below. To transmit, align the packet in the buffer so the last byte of the packet coincides with the last byte of the transmit buffer. Set up MEXHDR to contain the offset of the first byte of the packet. Finally, set TBSW to one. As long as TBSW remains one, the transmit buffer is busy and belongs to the controller; (any reference to it is ignored). When TBSW becomes zero, transmission is complete and the transmit buffer is available to software once again. **NOTE:** The TBSW bit

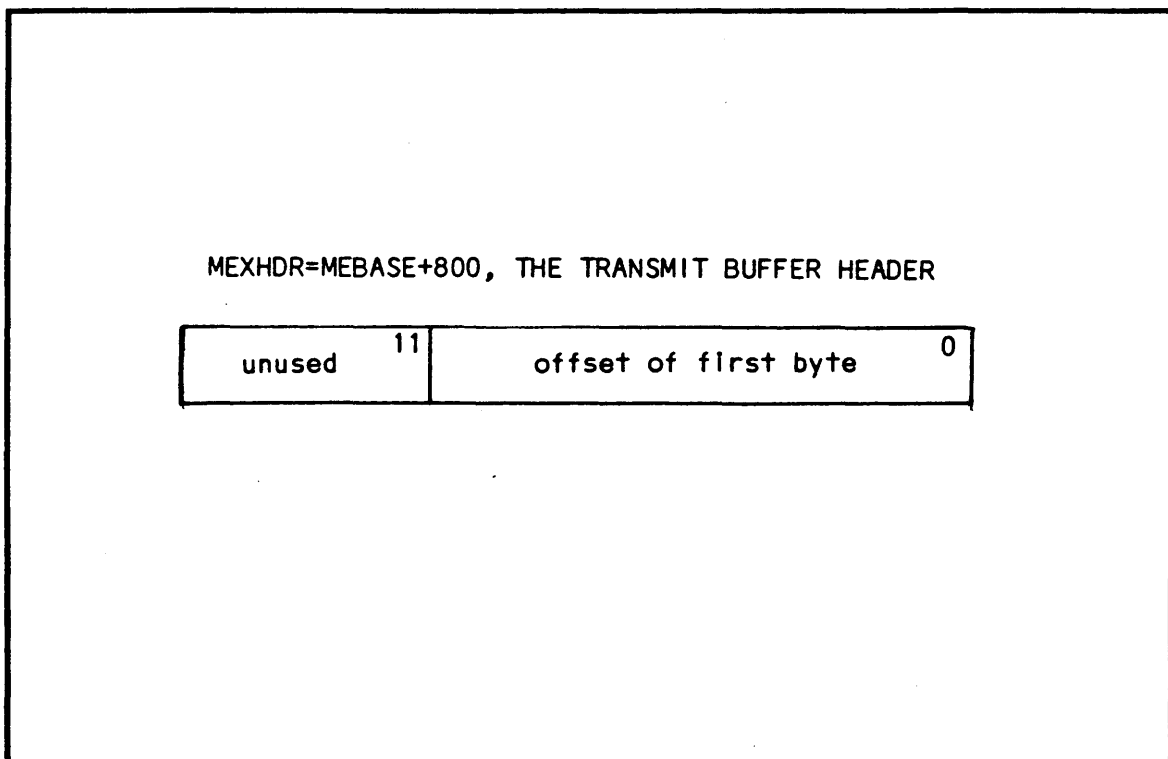


FIGURE 4-3. MEXHDR=MEBASE+800, THE TRANSMIT BUFFER

remains "1" during all JAM processing and will only become "0" again after the packet is successfully transmitted.

In the event of a collision, the controller sets JAM to one. To retransmit the packet, software must write the two's complement of the number of slot times to delay into MEBACK; then write a one back into JAM (the JAM bit is cleared by writing a one into it). Writing into MEBACK when JAM is not set produces unpredictable results.

The ME provides two buffers to receive packets from the ether. Software controls these buffers with two bits in MECSR; one for each buffer. ABSW controls receive buffer A which starts at MEBASE+1000H. BBSW controls receive buffer B which starts at MEBASE+1800H. (See Figure 4-1 on Page 4-1) To receive a packet in A, set ABSW to one. While ABSW remains a one, any reference to A will be ignored. ABSW will remain one as long as the buffer belongs to the controller; when ABSW becomes zero the buffer contains a packet and belongs to the software. To receive a packet in B, the software manipulates BBSW similarly. If both ABSW and BBSW are zero after giving both receive buffers to the controller, RBBA helps the software decide which buffer has the oldest packet: If RBBA is zero, then the packet in A is older than the packet in B. If RBBA is one the packet in B is older than the packet in A.

The header (first) word of both receive buffers are called, MEHDR

and MEBHDR, see Figure 4-4 below. The received packet minus the preamble starts immediately after the buffer header. After a packet arrives, the controller writes a status word in the buffer header. The low order eleven bits specify the offset of the first free byte after the packet. The high order bits contain various status flags, starting with the sign bit: fcs error, broadcast, range error, station address match, and framing error.

NOTE: In Figure 4-4 the "address match" status bit is "inverted". That is, the status bit is zero (0) if the destination address matches the station address, or one (1) if it does not match.

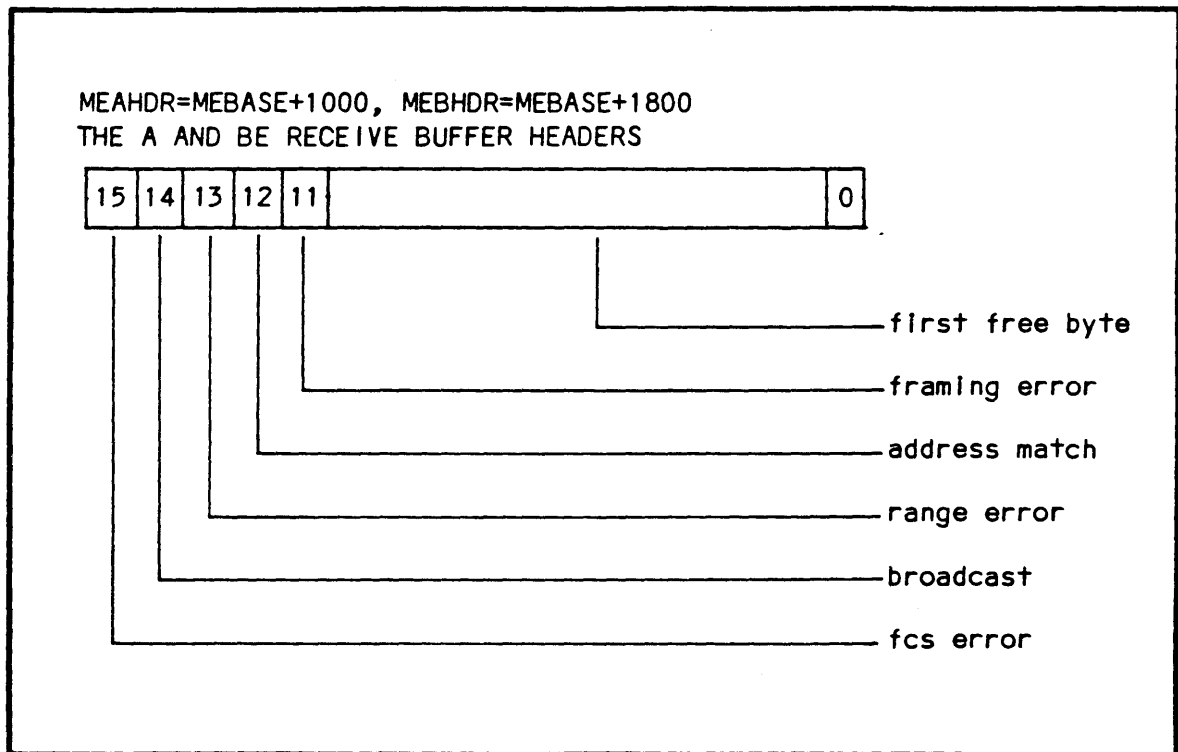


FIGURE 4-4. A AND B RECEIVE BUFFER HEADERS

Similarly, the "broadcast" status bit is zero (0) if the destination address matches the broadcast address, or one (1) if it doesn't match.

PA allows the software to select which packets the controller will accept on receive. The controller classifies all packets on the ether, as follows:

All	the universe of packets on the ether
Range	runts and oversized packets
FCS	packets with frame check sequence errors
Frame	packets with alignment errors
Mine	packets with destination address = station address
Multi	multicast packets <sup>1</sup>
Broad	broadcast packet <sup>2</sup>
Errors	range+fcs+frame

---

<sup>1</sup>NOTE: There is no status bit indicating a multicast packet that is not a broadcast packet. Such a packet is identified by a broadcast status bit equal to "1" (false) and the multicast bit in the destination address being "1" (true). Multicast packets will only appear in receive buffers if the receiver is enabled to accept them. The multicast bit of the destination address is the least significant (low order) bit of the first byte of the packet. In a receive buffer, this is the byte immediately following the buffer header word.

<sup>2</sup>NOTE: Broadcast is a special case of multicast. Enabling the receiver for multicast (modes 3,4,5) includes the broadcast address.

The software sets PA to receive only selected classes of packets.

- 0 all
- 1 all - errors
- 2 all - fcs - frame
- 3 mine + multi
- 4 mine + multi - errors
- 5 mine + multi - fcs - frame
- 6 mine + broad
- 7 mine + broad - errors
- 8 mine + broad - fcs - frame

Jumpers on the board select a single interrupt level for the ME controller. Setting JINTEN enables interrupts when JAM is set. Setting TINTEN enables interrupts when TBSW is zero. Setting AINTEN and BINTEN enables interrupts when ABSW and BBSW are zero respectively. The ME interrupts whenever any of the masked bits meet the conditions stated above; in particular, interrupts are not edge triggered, but level triggered. An interrupt routine should not leave an interrupt enabled unless it turns around the buffer that caused the interrupt. If the interrupt routine does not turn around the associated buffer and leaves the interrupt enabled, the controller will immediately reinterrupt as soon interrupt routine completes.

### 4.3 PROGRAMMING ANOMALIES

MECSR and MEBACK are replicated all through the control region. There is no method to read the value of MEBACK. If the software attempts to read MEBACK, it will read MECSR instead.

The station address ROM is the first six bytes of an eight byte block; this eight byte block is replicated throughout the region.

The station address RAM is the first six bytes of an eight byte block; this eight byte block is replicated throughout the region.

The interrupt enables for the transmit or receive buffers must NOT be set until that buffer has been assigned to the Ether. Enabling the interrupts before that time will result in an immediate interrupt going to the processor. Part of the interrupt service routine must include disabling the interrupt enables; otherwise, the processor will stay interrupted. The JAM interrupt can be left in the enabled state as long as desired.

**NOTE:**The packet stored into the packet buffer by the processor does not include the preamble or FCS, but does include the Ethernet data link layer header fields (destination address, source address, and type field) along with the data. Since the maximum legal packet size on the



Ethernet is 1514 (excluding the frame check sequence (FCS) field), at least 532 bytes of each transmit packet buffer will always be unused. Conversely, since the minimum legal packet size is 60 bytes (again excluding the FCS field), at most 1986 bytes of each transmit packet will be left unused. It is the responsibility of the driver software to insure that minimum and maximum packet size requirements are observed.

**4.4 OPERATION**

The transmitter and receiver operate independently, giving the programmer the illusion that the Ethernet is a full-duplex device. In fact, only one packet may exist on the coaxial cable at a time.

Whenever the Ethernet channel is not in use, the packet supplied to the transmitter is transmitted as quickly as possible. The ME hardware separates packets by the minimum packet spacing (9.6 - 10.2 microseconds).

Whenever a packet appears on the Ethernet, (except while transmitting) it is read into a buffer previously supplied to the receiver.

One aspect of the transmit process, the **binary exponential backoff** algorithm, is implemented by the ME hardware and requires software support only for random number generation. When the controller detects a collision it sets the JAM bit in the transmit control register which causes a 32-bit jam signal to be transmitted and, if the JINTEN bit is set, an interrupt occurs. The software must store a random number into the MEBACK register to cause a delay (back off) of the appropriate time after a collision. After the delay, the same packet is retransmitted. If a collision happens again, the cycle repeats.

The delay time is an integral multiple of a slot time (512 bit-times

or 51.2 microseconds). The integral multiple is chosen as a uniformly distributed random integer greater than or equal to zero and less than  $2^k$  (2 to the  $k^{\text{th}}$  power), where  $k$  is either the number of retransmission attempts for the packet being transmitted or 10, whichever is less. This algorithm doubles the mean of the delay time each time a collision occurs, ensuring the stability of the Ethernet even under extreme loading.

If the number of retransmission attempts exceeds 15 (probably a transmission subsystem malfunction), an error should be reported. The number of retransmissions normally exceeds 15 only when the Ethernet is broken. When this occurs, the only way to get the ME to return the transmit buffer to the processor is to reset the controller by setting bit 8 of MECSR. NOTE: This reset merely gives the memory buffers back to the Multibus (resetting TBSW, ABSW, and BBSW), reset does not affect the contents of any ME memories.

The processor overhead to support backoff in software is minimal. Studies show that Ethernet packets typically experience collisions less than 0.03% of the time.

**CHAPTER 5**  
**INSTALLATION AND CONFIGURATION CONSIDERATIONS**

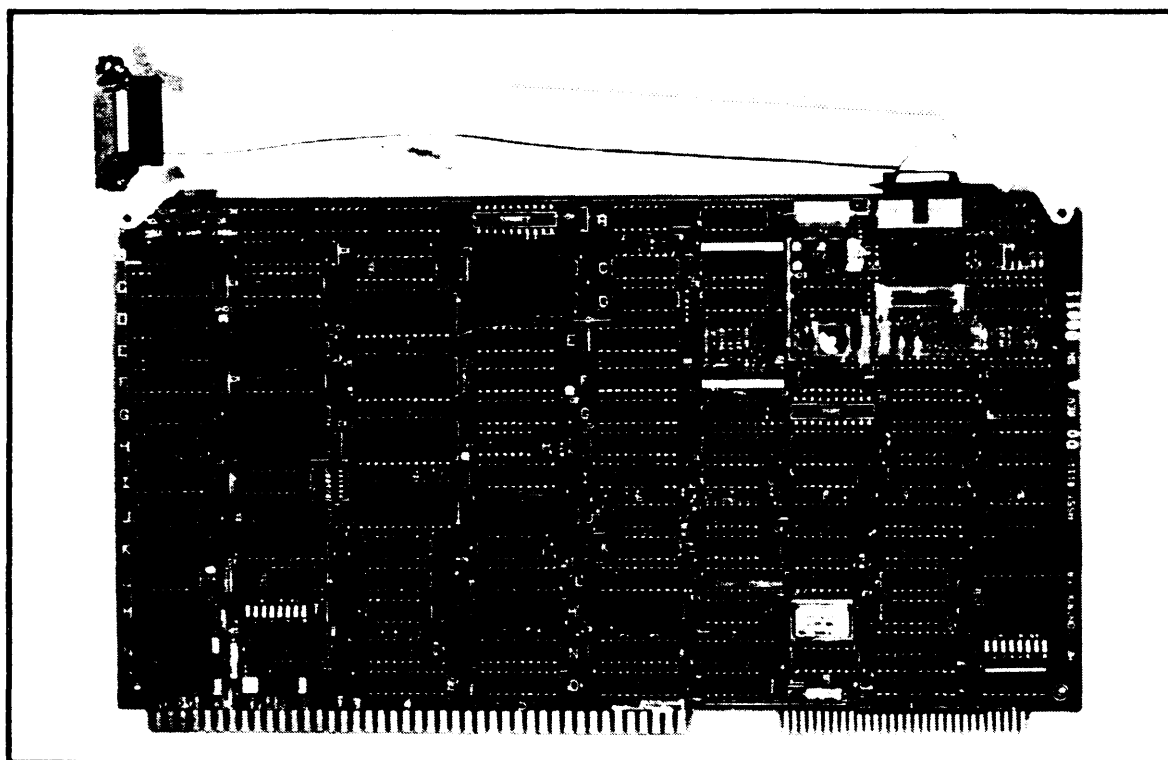
**5.1 INSTALLATION CHECKLIST**

- \_\_\_ **BEFORE OPENING THE SHIPPING CARTON, inspect it for damage or water stains, if so,**
  - \_\_\_ Write a brief description of the damage on the bill of lading, and
  - \_\_\_ Request that the carrier's agent be present when the carton is opened.
  - \_\_\_ Save the carton and packing materials to show the carrier in case the controller was damaged. The carrier is liable for shipping damage.
  
- \_\_\_ Unpack the ME Controller module gently by removing the foam packing material.

— Verify that all components are present. (See Figure 5-1 below.)

**One PC Board**

**One Cable**

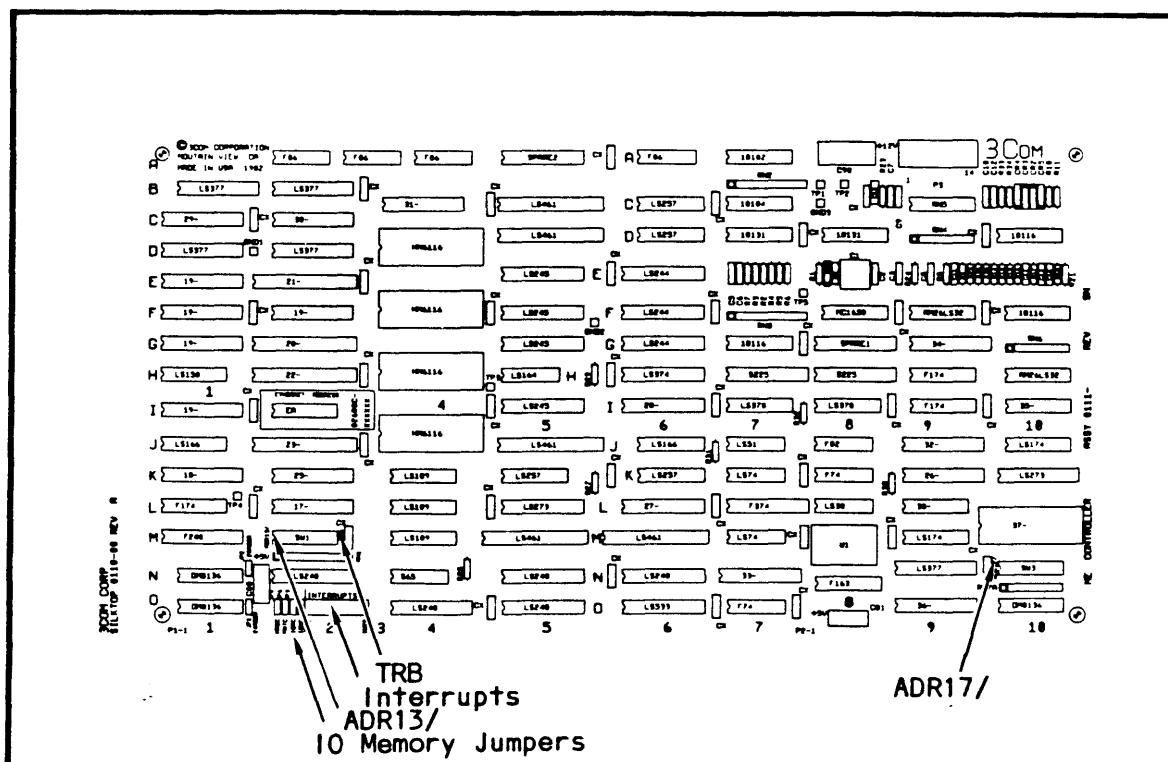


**FIGURE 5-1. ME CONTROLLER PARTS**

\_\_\_ Locate the byte ordering switch labeled TRB, see legend in Figure 5-2 below.

The byte ordering switch is configured at the factory to address high-order bytes first, like the Motorola 68000 microprocessor. If the ME Controller is going to be operated by a processor that addresses bytes in the reverse order (low-order first like the Intel 8086 microprocessor) turn the switch to the ON position.

\_\_\_ Locate I/O-Memory jumpers (labeled MRDC, MWTC, IORC, IOWC). (See Legend in Figure 5-2). To configure the ME Controller on the **memory side**, connect the memory jumpers labeled MRDC and MWTC. To



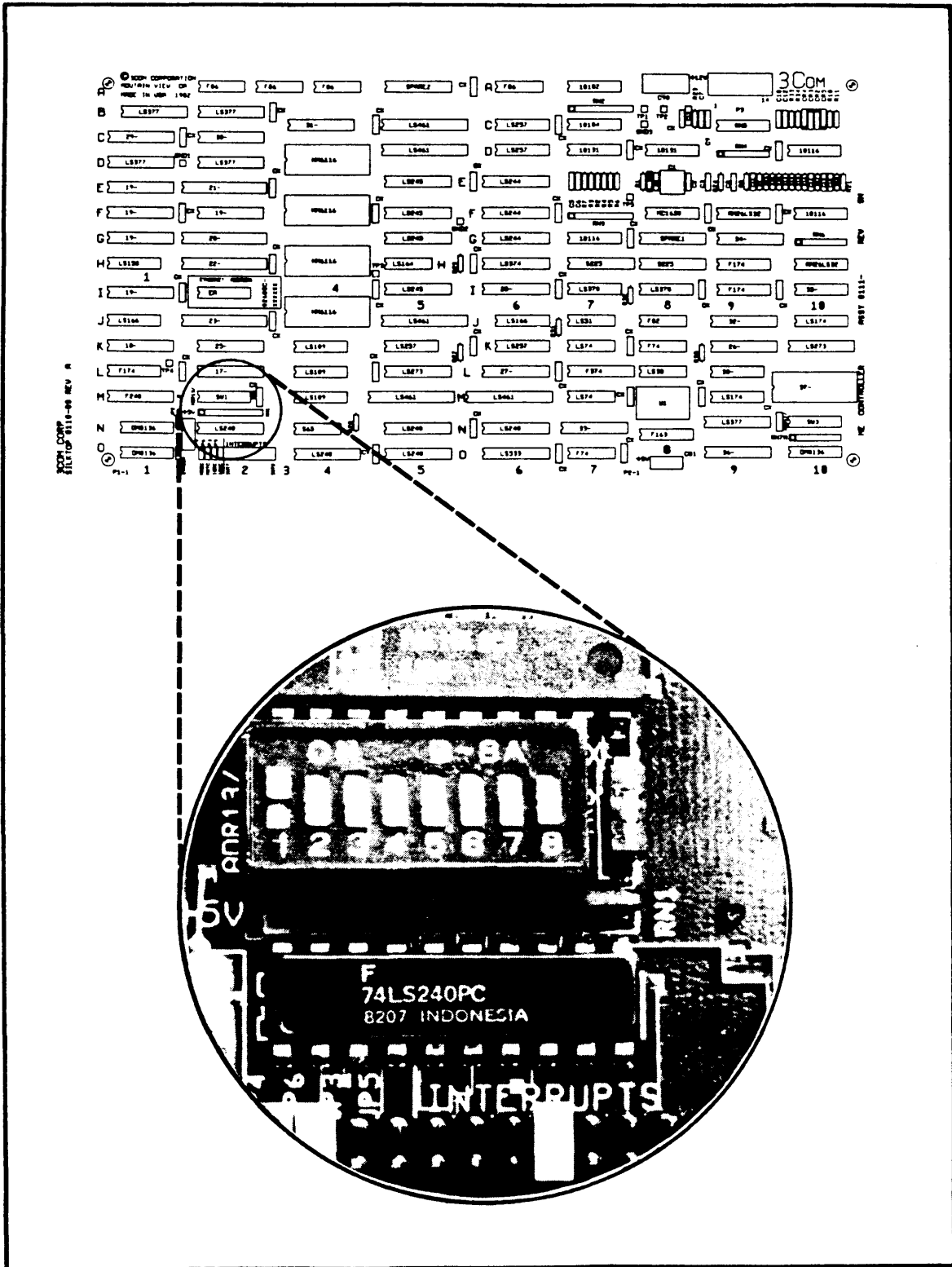
**FIGURE 5-2. ME CONTROLLER BOARD LEGEND**

configure the ME Controller on the **I/O side**, connect the I/O jumpers, IORC and IOWC.

\_\_\_ Locate the address switches labeled ADR13/ and ADR17/. (See **Figures 5-2, 5-3, 5-4**). The switches labeled ADR13/ specify the most significant bit for 20-bit addressing. Switch ADR17/ specifies the most significant bit for 24-bit addressing. If the factory default settings (**80000H**) are satisfactory, go the next step, otherwise change the switch settings to meet your specific needs.

\_\_\_ Locate the interrupt jumpers labeled INTERRUPTS. (See **Figures 5-2, 5-3**). Set the interrupt switches to the interrupt level desired: **From 0=highest, to 7=lowest.**

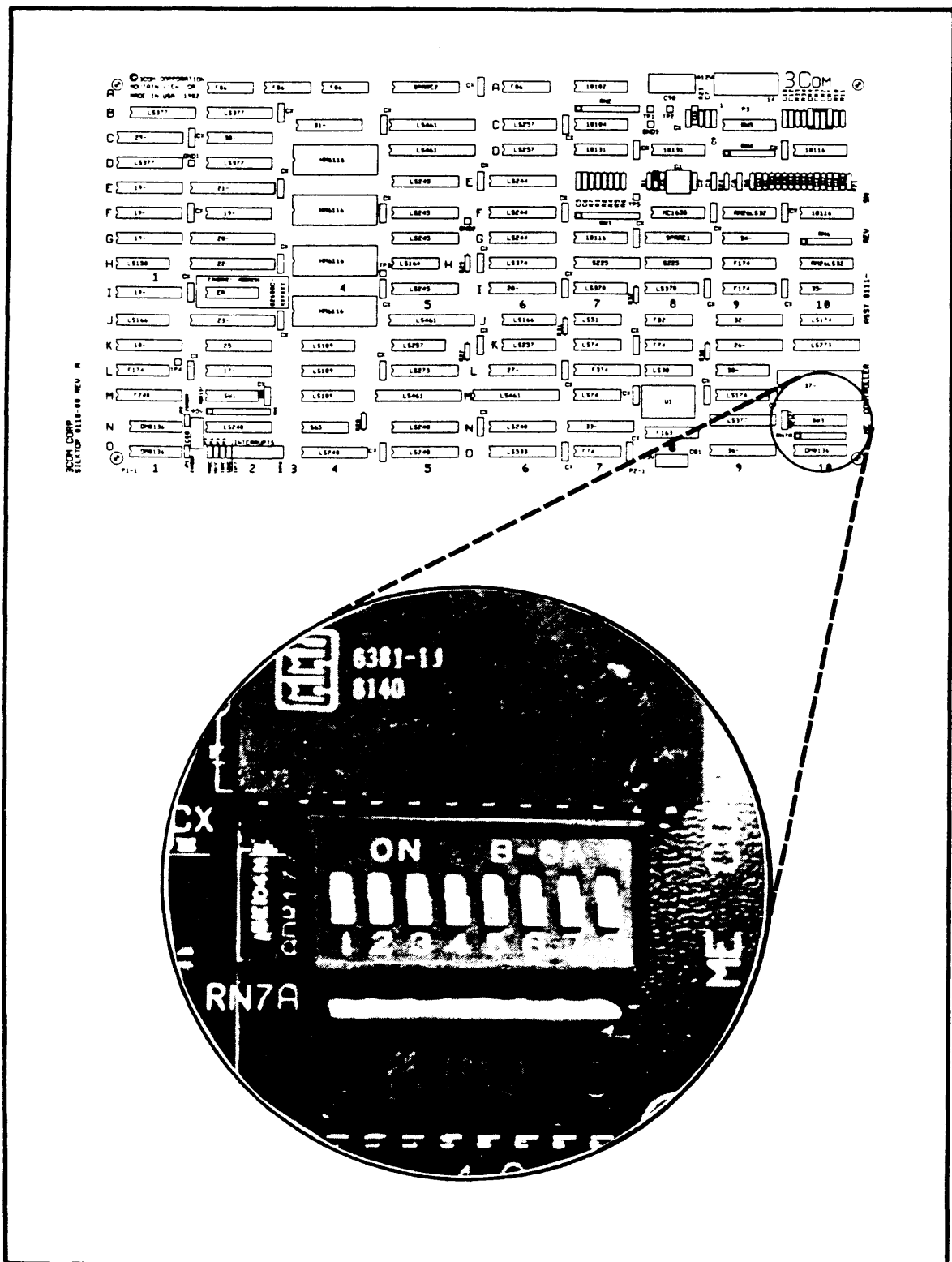
(Installation procedures continued after figures and tables on following pages.)



**FIGURE 5-3. ADDRESS SWITCHES ADR13/ TO ADR0D/ AND TRB SWITCH**

(Also see Figure 5-4.)





**FIGURE 5-4. ADDRESS SWITCHES ADR17/ TO ADR14/**

(Also see Figure 5-3.)

**TABLE OF SWITCH AND JUMPER SETTINGS**

<b>Description, Switches</b>	<b>Factory Default</b>	<b>Switch Name</b>
Buffer Memory Base Address	80000 H	Memory ADR17/-ADROD/
Byte ordering	OFF	TRB

(Most significant byte at EVEN address)

e.g. UNLIKE the 8086!

<b>Description, Jumpers</b>	<b>Factory Default</b>	<b>Jumper Name</b>
24-bit Addressing	Inserted	JP1
20-bit Addressing	Inserted	JP2

**Memory Configuration**

	<b>JP1</b>	<b>JP2</b>
24-bit Addressing	Inserted	Inserted
20-bit Addressing	Out	Inserted
16-bit Addressing	Out	Out

- \_\_\_ Turn the DC power to the backplane OFF.
  
- \_\_\_ Plug the serial cable supplied into the 14-pin connector on the back edge of the ME. The Notched Position of the Serial Cable 14-pin leader has "component-side" orientation.
  
- \_\_\_ Plug the Multibus board into the Multibus backplane.
  
- \_\_\_ Plug the Ethernet transceiver cable (not supplied) into the subminiature - 0 end of the serial cable.
  
- \_\_\_ Attach the lug at the end of the pigtail on the serial cable (near the board) to any convenient chassis ground. PROPER GROUNDING IS CRITICAL TO THE PROPER OPERATION OF THE ETHERNET.

## 5.2 ETHERNET ADDRESS CONSIDERATIONS

Each station on an Ethernet has a 48-bit Ethernet address associated with it that is unique across all Ethernets in the universe. The station address is assigned by the manufacturer of the station. For stations assembled from components from multiple manufacturers, the assignment of Ethernet addresses is ambiguous. This section describes the conventions to be followed to maintain unique Ethernet addresses under various configurations of components.

Each ME Controller manufactured by 3Com is shipped with a unique Ethernet address. The address is contained in a special address recognition PROM, and is also printed on the PC board in indelible ink. For stations that contain only a single ME Controller, the station address is the one supplied with the ME Controller. For stations that contain multiple ME Controllers, such as gateways, one of the controllers should be arbitrarily selected to contribute its Ethernet address as the station address. The hardware address recognition in the station must be programmed to respond to the chosen Ethernet address in all the ME Controllers attached.

It may become necessary in order to analyze a hardware problem to swap boards between stations at a user site. In that case it is the user's option either to have the Ethernet address to follow the board, or to disassociate the board from the address in software. The advantage of the former is that the address recognized always matches the address

printed on at least one of the controllers in the station. The advantage of the latter is the ability to swap hardware around without changing any station name/station address directory entries for the network.

If a board is sent back to the factory for repair, the board that comes back may not have the same Ethernet address printed on it. The customer has the option of using either the new Ethernet address or the old. Both addresses are allocated to the customer and are under the customer's control.

It is the customer's responsibility to manage the allocated Ethernet addresses. Addresses can be assigned in any manner as long as none are duplicated on more than one station. The recommended practice is to add each ME Controller Ethernet address to a site-wide pool of addresses maintained independently of the hardware. That way, ME hardware can be moved around among stations or sent back to 3Com for repair without affecting any software.

**CHAPTER 6**

**FACTORY WARRANTY AND REPAIR POLICY**

Any 3Com product which fails within 90 days from date of shipment will be repaired or replaced by 3Com free of charge providing, in the opinion of 3Com, the product has not been subject to improper electrical, mechanical, or thermal stress. The customer should send the defective item to 3Com stating the nature of the fault and quoting the date and invoice number of the original shipment. Freight charges from customer to 3Com must be paid by the customer. Freight charges from 3Com to U.S. customers are paid by 3Com using UPS. Foreign customers must pay freight both ways. Replacement or repair under warranty will normally be completed within seven days of receipt at 3Com.

After 90 days, a factory repair service is available for the 3C400 MULTIBUS Ethernet Controller. The charge is fixed at \$175 for each controller regardless of the extent of the repairs concerned. If testing of the controller shows no failure, or if the controller has been modified or damaged by the customer attempting repairs or if it has been subject to improper electrical, mechanical, or thermal stress, it will be returned to the customer in the same condition as received and a \$100 charge assessed. Controllers repaired under the fixed-price repair service will be warranted for 90 days from the date of shipment of the repaired controller.

In order to achieve a quick turnaround, 3Com may ship a controller different from that received. In all cases, the controller shipped from 3Com will be given a new Ethernet address to avoid any possible duplication of addresses. The customer may use the Ethernet address originally shipped with the controller, or with the repaired controller, or both.

**APPENDIX A**  
**ORDERING INFORMATION**

<b>Model Number</b>	<b>Description</b>
3C100	Ethernet transceiver
3C110-015	Transceiver cable, 15 meter
3C400	Multibus Ethernet Controller
3C420	Reference Manual
3C440	Multibus Ethernet Starter Package

**Warranty**

Any item which fails within 90 days of shipment will be repaired or replaced by 3Com free of charge. After 90 days a billable factory repair service is available.



**APPENDIX B**  
**BIBLIOGRAPHY**

1. DEC-INTEL-XEROX, "The Ethernet, A Local Computer Network, Data Link Layer and Physical Layer, Version 1.0", September 30, 1980.
2. Robert M. Metcalfe and David R Boggs, "Ethernet: A Distributed Packet Switching for Local Computer Networks", CACM, 19:17, July 1976, pp. 395 - 404.
3. John F. Shoch and Jon A. Hupp, "Performance of an Ethernet Local Network - A Preliminary Report", Local Area Communications Network Symposium, Mitre and NBS, Boston, May 1979.