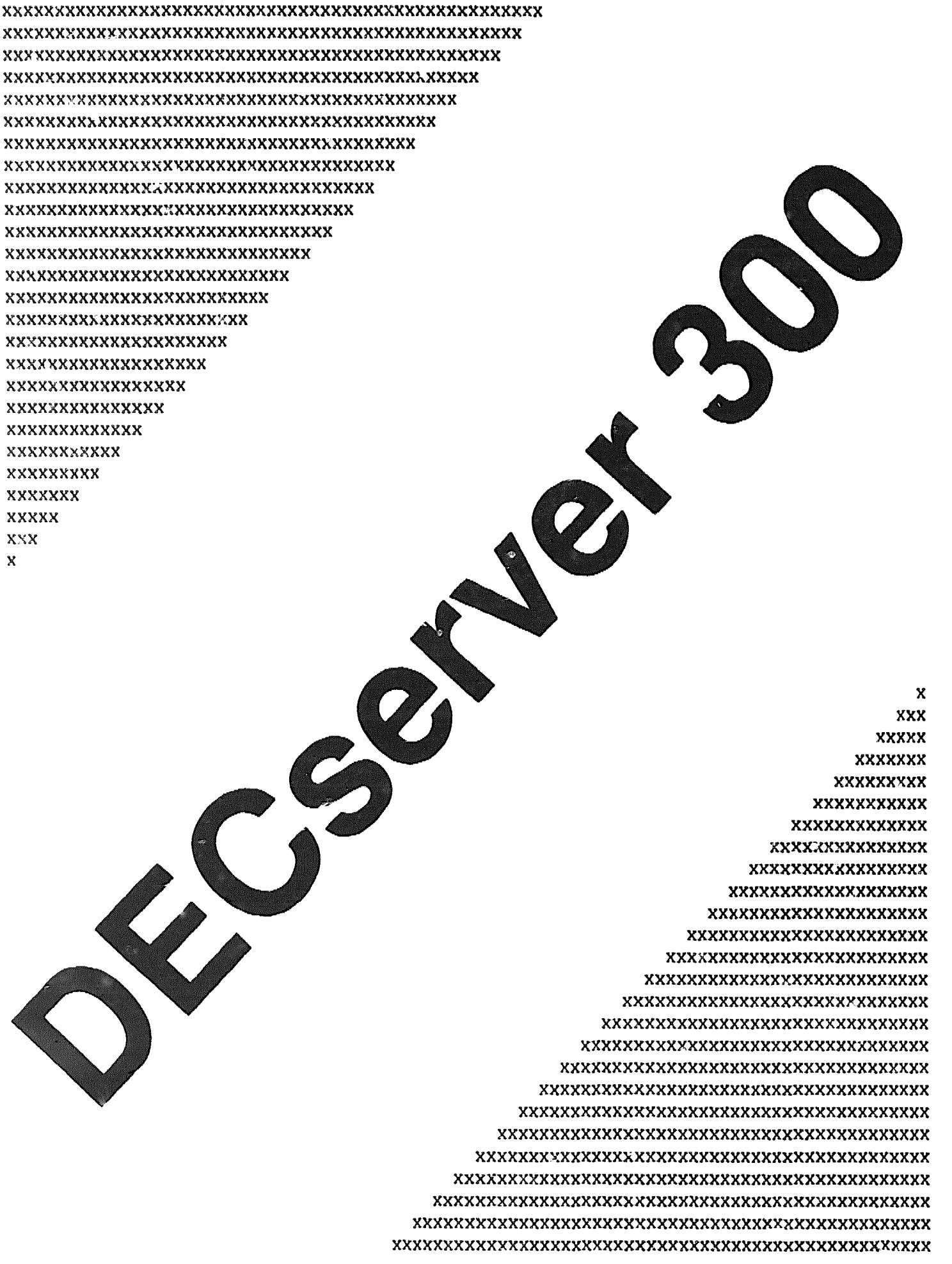

DECserver 300

DECserver 300

Technical Description

August 1989

This manual describes how the DECserver 300 system software, firmware, and hardware interact to perform server functions. The manual also describes the hardware specifications, controls, indicators and diagnostics. This manual is an aid to training, field service, and manufacturing personnel.

Supersession/Update Information: This is a new manual.

digital™

Order Number: EK-A0367-TM-001

EK–A0367–TM–001
August 1989

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment assumes no responsibility for any errors that may appear in this document.

Copyright © 1989 by Digital Equipment Corporation
All Rights Reserved

The postage–prepaid Reader's Comments form on the last page of this document requests the user's critical evaluation to assist us in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

DEBNA	DESVA	Q–Bus
DEBNET	DEUNA	Rainbow
DEC	digital™	RSX
DECconnect	LAN Bridge	RSX–11M–PLUS
DECnet	LA120	ThinWire
DECserver	LN01	ULTRIX–32
DECUS	LN03	UNIBUS
DELNI	LQP02	VAX
DELUA	MASSBUS	VAXcluster
DELQA	Micro/RSX	VAXmate
DEMPR	MicroVAX	VAXstation
DEQNA	PDP	VMS
DESOA	Professional	

AT&T is a trademark of American Telephone and Telegraph Company.
IBM is a registered trademark of International Business Machines Corporation.
PC/AT and PC/XT are trademarks of International Business Machines Corporation.

This manual was produced by Networks and Communications Publications.

Contents

3 Server Hardware Functional Description

4 Software Functional Operation

5 Firmware Functional Operation

Figures

Tables

PAGE viii INTENTIONALLY LEFT BLANK

Preface

This manual provides a functional description of system components and describes how the software and hardware components interact to perform server functions. It also lists hardware specifications and cables and describes read-only memory (ROM) based firmware.

Intended Audience

The *DECserver 300 Technical Description* is intended as a resource for training, field service, and manufacturing personnel. A basic knowledge of Ethernet local area networks is a prerequisite to understanding this manual.

Document Structure

Chapter 1	Introduces local area transport networks and provides an overview of the DECserver 300 hardware, software, and ROM-based firmware.
Chapter 2	Describes DECserver 300 hardware specifications, cables, connectors, controls, and lists supported devices.
Chapter 3	Describes DECserver 300 hardware functions.
Chapter 4	Describes DECserver 300 software functions
Chapter 5	Describes DECserver 300 firmware functions including self-test diagnostics.
Appendix A	Describes the DECserver 300 maintenance operation protocol (MOP) system ID format.

Other DECserver 300 Manuals

- *DECserver 300 Introduction*

  Outlines the DECserver 300 system, hardware and software, and provides information for using, expanding or reconfiguring the server. Describes the documentation and provides flowcharts for reading sequences for different audiences. This document is intended for any user of the DECserver 300.

- *DECserver 300 Hardware Installation*

  Describes the DECserver 300 hardware and explains how to install it. This manual is intended for the hardware installer and the server manager.

- *DECserver 300 Identification Card*

  Provides the space to record the serial number, Ethernet address, DECnet node address, and DECnet node name of the server. This document is intended for the network manager, the software installer, and the server manager.

- *DECserver 300 Software Installation (op-sys)*

  Explains how to install the DECserver 300 distribution software, how to configure down-line load hosts, and how to verify the DECserver 300 system installation. In the title, (op-sys) is the name of the load host operating system. This guide is intended for the load host system manager and the server manager.

- *DECserver 300 Use*

  Describes the user interface and the general functions of the server. This guide provides complete information for using all nonprivileged server commands. This guide is intended for users of interactive terminals connected to DECserver 300 ports.

- *Terminal Server User's Reference Card*

  Describes and gives examples of the most frequently used nonprivileged server commands on a reference card. This card is intended for users of interactive terminals connected to server ports.

- *DECserver 300 Management*

  Describes all the initial and day-to-day management tasks required to the DECserver 300 manager. The topics cover all the information needed to configure the ports and to customize the permanent and operational databases of the server. This guide is intended to the DECserver 300 manager.

- *Terminal Server Commands and Messages Reference*

  Describes the usage and syntax of all terminal server commands. This reference also lists and describes all status and error messages issued by the server. This reference is intended for the server manager but is useful for terminal users who want more detailed reference information.

- *Local Area Transport (LAT) Network Concepts*

  Describes the local area transport (LAT) protocol, and LAT network concepts. This document is intended for the server manager, the system manager, and the network manager.

- *Terminal Server Glossary*

  Defines terms used in the server documentation sets. This is intended as a reference tool for all users of server documentation.

DECserver 300 On-Line Documentation

- *DECserver 300 Release Notes*

  Describes any discrepancies between the actual product and the information in the documentation set. These notes are intended for the software installer and the DECserver 300 manager.

- *On-Line Help*

  Provides two forms of server help: tutorial help and command reference help. Tutorial help provides basic information about logging in and using the server. Command reference help provides detailed information about using all the server commands available at your privilege level. Help is intended for all server users.

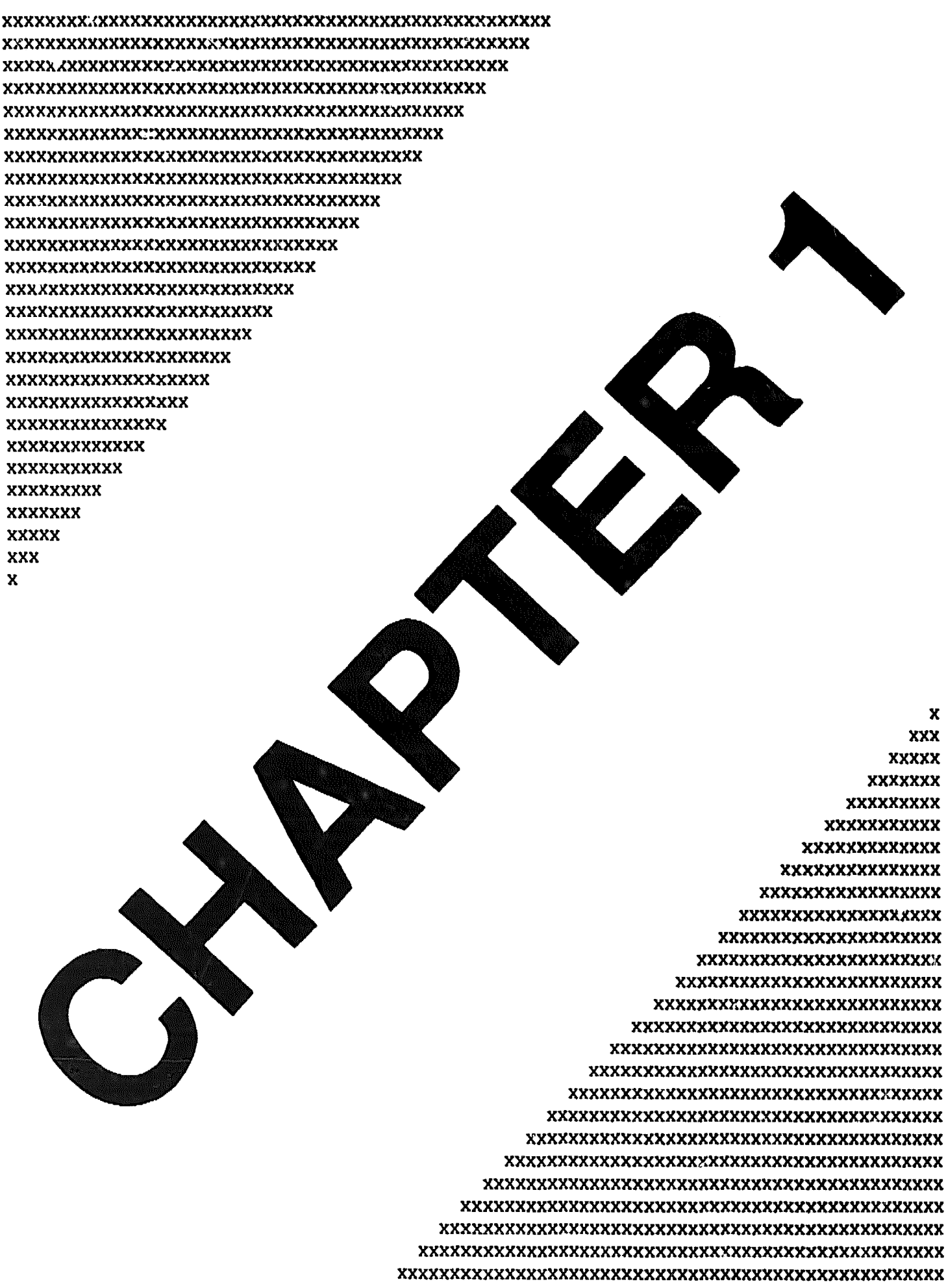
Associated Documents

- *Guide to Terminal Server Manager* and *Terminal Server Manager Software Installation Guide*

  These documents contain the information necessary to install and run the Terminal Server Manager (TSM) software, an optional network management product, which is installed onto a VAX/VMS system running DECnet-VAX. These guides describe how to use TSM to manage a mix of Digital Equipment Corporation Ethernet terminal servers connected for the installer and manager of the TSM software product.

- *Support Print Set* (Digital internal use only)

  Contains circuit schematics, unit assembly drawings and parts list.


CHAPTER 1

1

System Overview

1.1 Introduction

The DECserver 300 server is a terminal server that enables devices to access and share the resources of host computers and other devices connected to the same Ethernet local area network (LAN). The devices supported include terminals, printers, and personal computers.

The DECserver 300 hardware supports the EIA-423-A electrical interface standard. EIA-423-A is compatible with the EIA-232-D interface, but supports longer cabling runs and higher signaling speeds. In addition to EIA-423-A, the DECserver 300 has used special protection components that minimize damage to computing equipment caused by electrical overstress (EOS) and electrostatic discharges (ESD). EOS and ESD are the most common causes of communications equipment failure.

The DECserver 300 server supports devices that require data-leads-only for operation. It has two additional signals, DSR and DTR, which may be used as follows:

- DSR logout — Logs out an attached device on power-down
- DSR/DTR — Flow control for printers or other attached devices
- DSR — Status signal check for printers

NOTE

The DECserver 300 server does not support connections to wide-area networks via modems, or connections to non-LAT hosts. Users requiring these capabilities should use the DECserver 200/MC which provides asynchronous modem support (both dial-in and dial-out), and uses the EIA-232-D interface.

1.1.1 Port Device Interface

The DECserver 300 server supports 16 asynchronous serial data communication channels which allows any combination of 16 attached devices access to the resources on a LAN via the 16 port device connectors. The port device connectors are modified modular jacks (MMJs).

1.1.2 Network Interfaces

The DECserver supports both ThinWire and standard Ethernet.

A transceiver cable connects the server to a standard Ethernet LAN. The transceiver may be connected to any of the following:

- Another transceiver cable section. This cable can be secured in an Etherjack junction box.
- A DELNI local area interconnect.
- A transceiver on a standard Ethernet coaxial cable for Digital Equipment Corporation baseband networks or a DECOM for Digital Equipment Corporation broadband networks.

A ThinWire coaxial cable connects the server to a ThinWire Ethernet LAN. The server may be connected to any of the following:

- A stand-alone ThinWire Ethernet coaxial segment
- A DEMPR ThinWire Ethernet coaxial segment
- A DESPR ThinWire Ethernet coaxial segment

1.1.3 Benefits of Terminal Servers

Connecting terminals remotely over a LAN to computer resources instead of directly to a single host computer has many benefits. Such as:

- Character interrupt handling is handled by the server instead of the host computer thus reducing the host computer overhead.
- From an interactive terminal, a user can connect to and use the resources of any service node on the LAN that supports the LAT protocol. Once a user has logged into the service node, he/she can use application programs and utilities as if connected directly to that node through a port device.

- Once connected to a service node, a user can suspend a current interactive session, open another session with the same node service or a different one, use a different service, and then suspend the second session and return to the first session picking up where he/she left off. While server users are normally limited to two or four sessions, the server can be configured to support up to eight sessions at a particular port. The multiple session capability saves the time normally required to return to the same point in the session.

- A user is not solely dependent on the availability of a single host computer. If the service node currently being used goes off line for any reason, a user can simply access a different service node and obtain the same service. If the server is connected to a VAX cluster, the session will automatically be switched to another service node in the cluster that offers the same service.

- The installation and management of systems and port devices have less impact on users (especially where users and computer systems are often added to moved to other physical locations on the same LAN).

- The procedure for configuring and expanding the system capabilities is simplified. Thus terminal servers further enhance the flexibility inherent in Digital Equipment Corporation extended LAN architecture.

1.2 Overview of Local Area Transport (LAT) Networks

A local area transport (LAT) network consists of LAT nodes (nodes running LAT protocol software), their network interfaces, and the LAN hardware (Ethernet cable and so forth) that connects these nodes.

The LAT protocol software processes communications directly over the LAN. The LAT protocol has its own transport mechanism and does not use the DECnet transport mechanism or the DECnet protocols and co-exists with other protocols on the same LAN. For example, a LAT network may also be a DECnet network, if some of the nodes on the LAN use DECnet protocols. Many DECnet nodes support LAT and DECnet simultaneously through the same physical Ethernet interface.

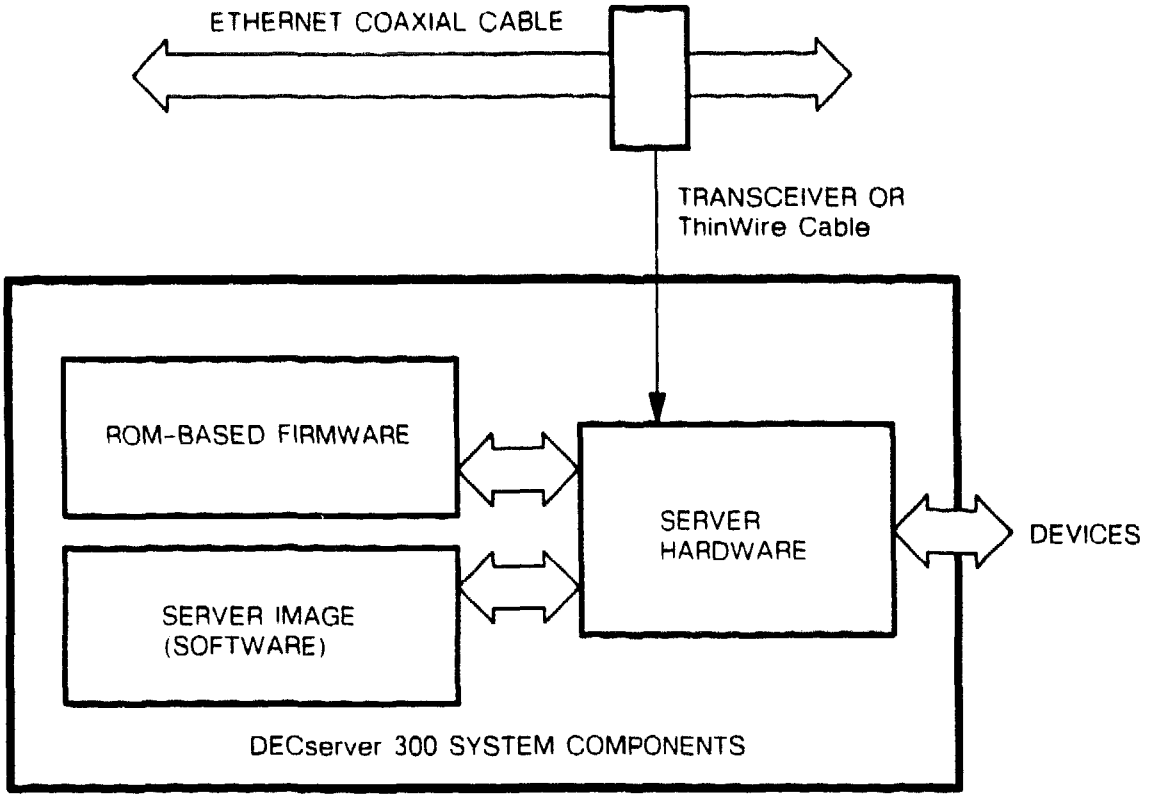
The only feature that makes any network node a LAT node is the presence of software that implements the LAT protocol. LAT software must reside on the servers and on the service nodes that interact with them. The LAT software that resides on nodes that offer services is referred to as service software.

1.3 Overview of the Server

The server comprises the following major components (see Figure 1-1):

- Hardware
- ROM-based firmware
- Server image (software)

Figure 1-1: DECserver 300 System Block Diagram



1.3.1 Server Hardware

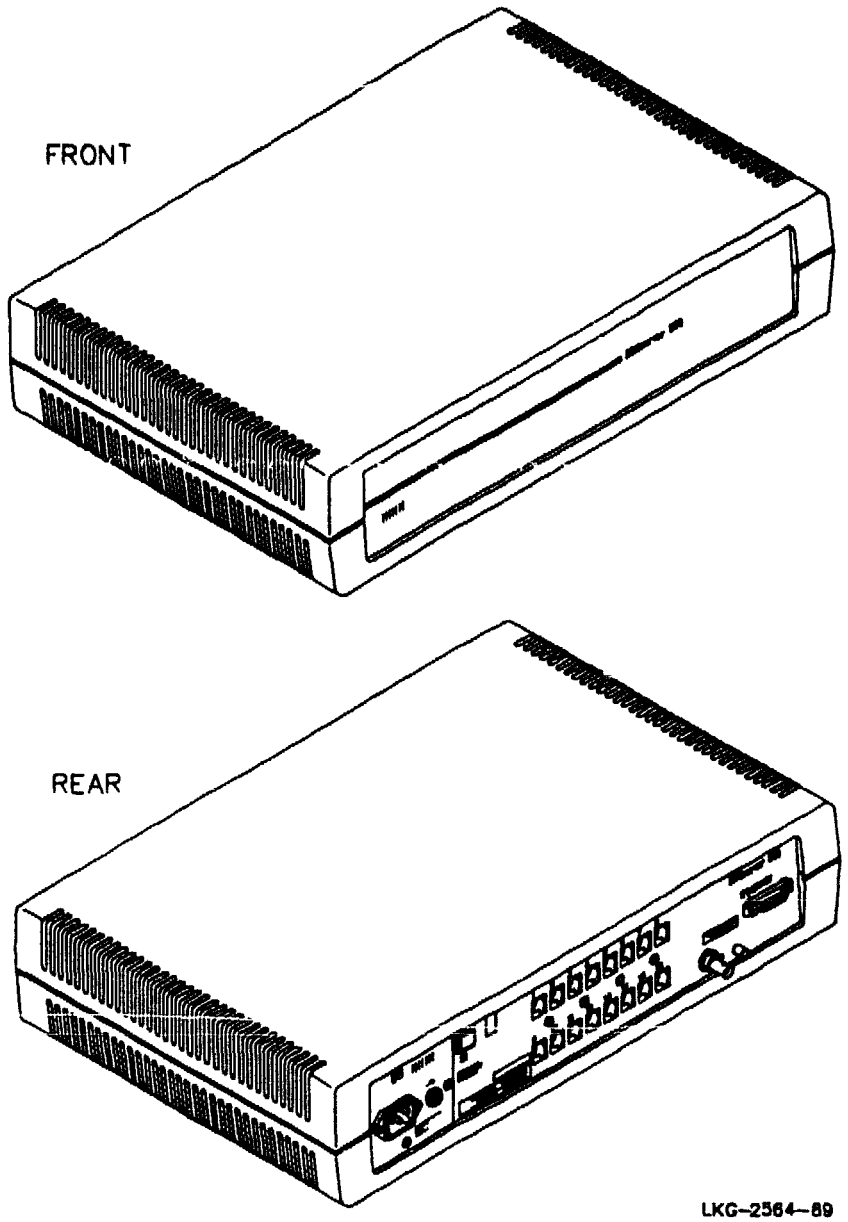
There is only one model of the DECserver 300 (DSRVF-Bx) but this model has two versions depending on the input voltage as shown Table 1-1:

Table 1-1: Versions of the DECserver 300 (DSRVF-Bx)

Model Version	Input Voltage
DSRVF-BA	100-120 Vac
DSRVF-BB	220-240 Vac

The server is shown in Figure 1–2.

Figure 1–2: DECserver 300 Model DSRVF-Bx



1.3.2 ROM-Based Firmware

The firmware is permanently stored in the ROM. The type of ROM used is electrically programmable read-only memory (EPROM). The firmware provides the following services:

- A diagnostic self-test program which automatically tests the following:
  - Internal hardware components
  - Interface to the standard and ThinWire Ethernet
  - Device port interfaces
- A maintenance operation protocol (MOP) program – The MOP program initiates down-line loads of the server image and, in the event of server software failures, initiates up-line dumps of server memory to the load host.
- Repair/debug utilities – The repair/debug utilities enable design, manufacturing, and repair personnel to locate faults.

1.3.3 Server Software

The server uses DECserver 300 software, Version 1.0. The software resides on the load host and comprises three components:

- The server image
- The load host configuration procedure: DSVCONFIG
- The Terminal Server Manager (TSM), version 1.3 or later — Optional

The load host can be any host computer on the same LAN that is running DECnet software, Phase IV and the appropriate operating system software. Refer to the *DECserver 300 Software Product Description* for information on the supported operating systems.

1.3.3.1 Server Image

The server image is the software which operates the server and must be down-line loaded from the load host.

Until the server image is loaded, the server can only run its ROM-based firmware. Once the server image is loaded, the server accesses the permanent database that is stored in the non-volatile random access memory (NVRAM) and creates an operational database in dynamic random access memory (DRAM). The type of non-volatile memory random access memory (NVRAM), used in the server, is electrically erasable programmable read only memory (EEPROM). The values in the operational database are used by the server image when operating the server.

NOTE

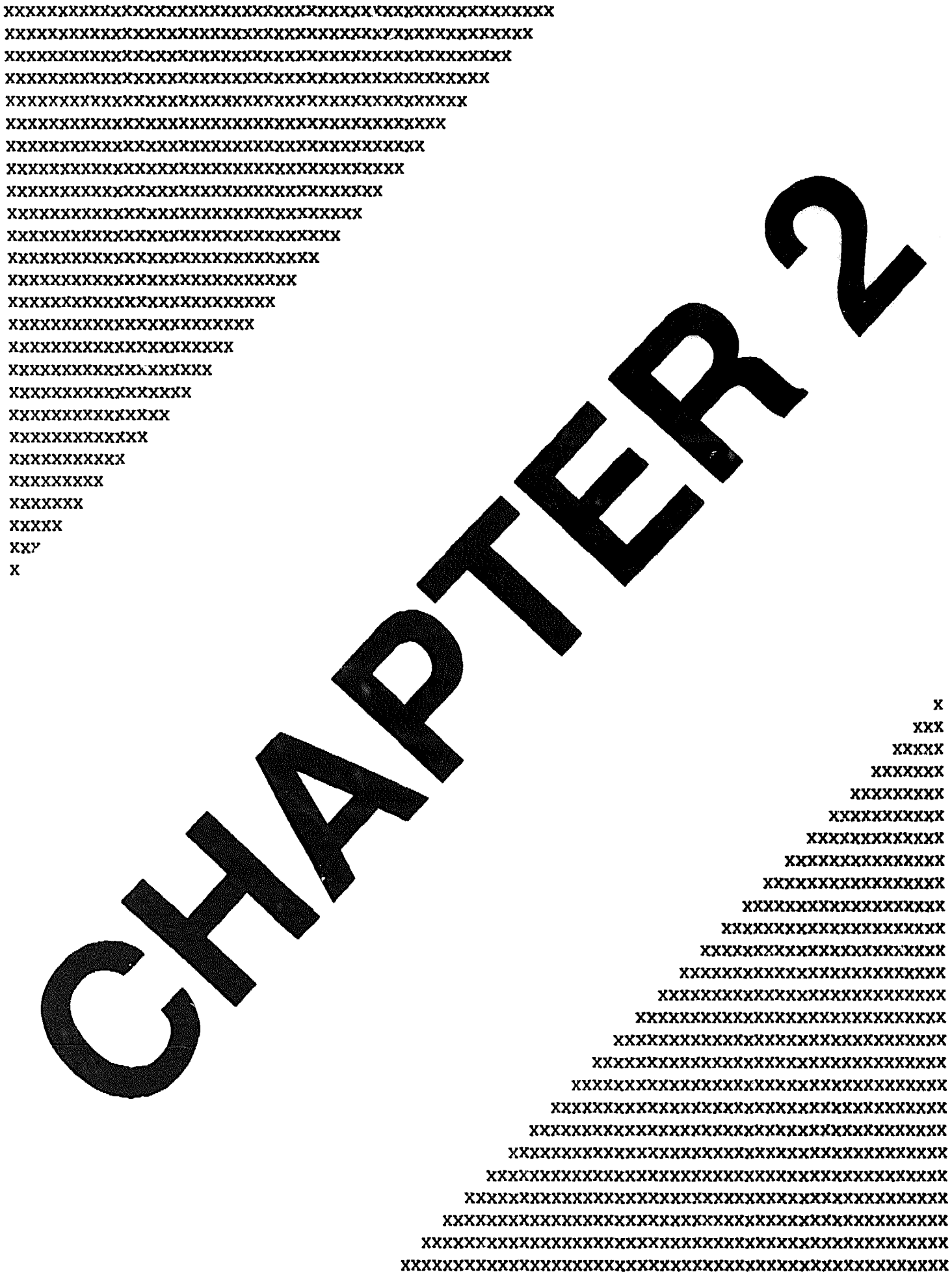
The operational database values can be changed by using the SET command. The permanent database values can be changed by using the DEFINE command. Refer to the *DECserver 300 Management* manual for further information on the server databases and procedures for changing them.

1.3.3.2 The Load Host Configuration Procedure: DSVCONFIG

The load host procedure (DSVCONFIG) is a command procedure which you use to configure your load host's database. The load host's database includes information on the servers available and their DECnet characteristics, which is necessary for down-line load and up-line dump. For further information on DSVCONFIG, see the *DECserver 300 Management* manual.

1.3.3.3 The Terminal Server Manager (TSM)

The Terminal Server Manager (TSM) is an optional software product that helps the server manager remotely monitor and control multiple servers on the LAN. TSM runs on suitably configured VMS systems running DECnet Phase IV.


CHAPTER 2

2

Server Hardware, Specifications, and Cables

2.1 Introduction

This chapter describes the server hardware, specifications, and options. Also, it lists the cables required to connect the server to the various port devices and to the Ethernet LAN.

2.2 Server Hardware

The following sections describe the server controls, indicators, connectors, and accessories that ship with the server

2.2.1 Server Controls, Indicators and Connectors

All of the server controls, indicators, and connectors that are used during server operation are located on the rear of the server and are shown in Figure 2-1. The server controls are described in Table 2-1. The indicators are described in Table 2-2 and a brief description of the connectors is given in Table 2-3.

Figure 2-1: DECserver 300 Back Panel

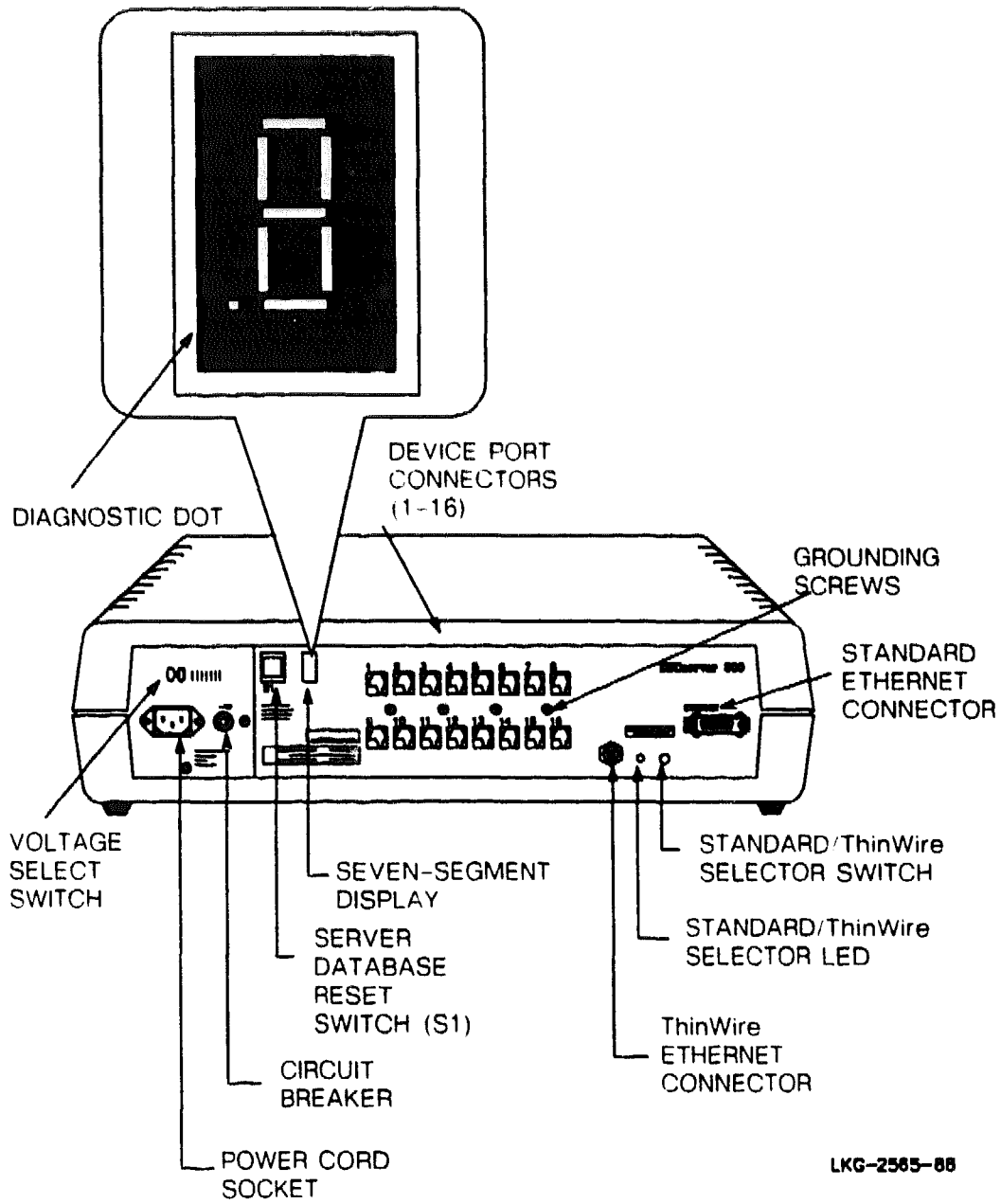


Table 2-1: Server Controls

Control	Description
Server database reset switch (S1)	This switch, in conjunction with power-up, restores factory-set parameters in the server's database.
Voltage select switch	This switch sets the input voltage to the range required (100-120 Vac or 220-240 Vac).
Circuit breaker	This circuit breaker (press to reset) protects the power supply against excessive current.
Standard/ThinWire selector switch	This switch selects either ThinWire or Standard Ethernet.

Table 2-2: Server Indicators

Indicator	Display
Standard/ThinWire selector LED	Lights (green) to indicate that the ThinWire Ethernet connector is selected.
Seven-Segment Display	This display provides error and status information.
Diagnostic Dot	The decimal point on the seven-segment display provides diagnostic information.

Table 2-3: Server Connectors

Connector	Description
Port device connectors	These are 16, 6-pin female modified modular jack (MMJ) connectors used to connect devices to the server.
Standard Ethernet connector	This single 15-pin female D-connector is used to connect to a standard Ethernet local area network using transceiver cable.
ThinWire Ethernet connector	This single female BNC connector is used to connect to a ThinWire Ethernet local area network using ThinWire cable and a T-connector.
Power cord receptacle	The server power cord plugs into this receptacle.
Grounding screws	These screws provide grounding points for shielded cables.

2.2.2 Server Connectors

This section describes the following DECserver 300 hardware connectors:

- Ethernet transceiver interface
- Device port connector — modified modular jack (MMJ)

2.2.2.1 Ethernet Transceiver Interface

The DECserver 300 hardware has a ThinWire Ethernet connector and a standard Ethernet connector. The ThinWire Ethernet connector is a 50-ohm, RG58-type, BNC connector with one pin and a shield.

The standard Ethernet transceiver interface matches the signal specifications described in *The Ethernet; A Local Area Network; Data Link Layer and Physical Layer Specification*.

Figure 2-2 shows how the pins are numbered on a standard Ethernet transceiver interface connector. The signals for the standard Ethernet connector pins are listed in Table 2-4.

Figure 2-2: Pin Numbers for the Standard Ethernet Transceiver Interface Connector

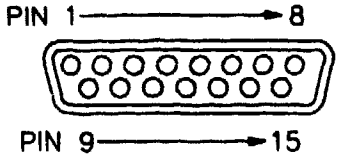


Table 2–4: Pin Descriptions for Standard Ethernet Transceiver Interface

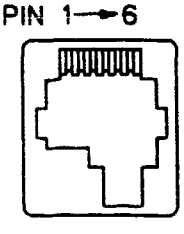
Pin Number	Signal Name
1	Shield
2	Collision presence +
3	Transmit +
4	Reserved
5	Receive +
6	+12 Volt power return
7	Reserved
8	Reserved
9	Collision presence -
10	Transmit -
11	Reserved
12	Receive -
13	+12 Volt power
14	Reserved
15	Reserved

2.2.2.2 Device Port Connectors

Modified modular jack (MMJ) connectors are used for connecting devices to the ports on the DECserver 300 hardware.

Figure 2–3 shows shows how the pins are numbered on an MMJ connector and Table 2–5 lists the signals on the pins.

Figure 2–3: Pin Numbers for the Device Port Connector



UKG-2583-89

Table 2-5: Pin Descriptions for Device Port Connector

Pin Number	Signal Name
1	DTR
2	Transmit data
3	Transmit common
4	Receive common
5	Receive data
6	DSR

2.2.3 DECserver 300 Accessories

The DECserver 300 ships with the following accessories:

- BNC T-connector (12-25869-01)
- BNC 50-ohm terminators (2) (12-26318-01)
- Software license
- H3103 loopback connector (12-25083-01)
- Country kit (must be ordered separately for DSRVF-BB)

2.2.3.1 DECserver 300 Country Kits

The DECserver 300 country kits consist of:

- Power cord
- *DECserver 300 Hardware Installation* manual
- *DECserver 300 Identification Card*
- Blank ID labels
- Rack mount kit
- FTZ card (Austria and Germany only)

2.3 Specifications

The following sections list server specifications

2.3.1 Physical Requirements

The DECserver 300 hardware should be placed at least 45 centimeters (18 inches) above the floor. Allow for 15 centimeters (6 inches) of airspace around the server air vents. Table 2–6 shows the size and weight of the server.

Table 2–6: Physical Specification

Dimension	Measurement
Width	49.3 cm (19.4 inches)
Height	11.7 cm (4.6 inches)
Depth	31.2 cm (12.3 inches)
Weight	5.4 kg (11.9 lbs)

2.3.2 Environmental Requirements

Environmental requirements for temperature and humidity must be within the ranges shown in Table 2–7.

Table 2–7: Environmental Specification

Parameter	Minimum	Maximum
Temperature †		
Operating	5°C (41°F)	50°C (122°F)
Nonoperating	-40°C (-40°F)	66°C (151°F)
Maximum temperature changes per hour	-	20°C (36°F)
Altitude	-	
Operating		2438 meters (8000 feet)
Nonoperating		4877 meters (16000 feet)
Relative Humidity		
Operating (noncondensing)	10%	95%
Nonoperating (noncondensing)	10%	95%

† If you are operating the server above 2.4 kilometers (8000 feet, decrease the operating temperature specification by 1.8°C for each 1000 meters (1°F for each 1000 feet).

2.3.3 Electrical Requirements

The Power at the electrical outlet must match the requirements shown in Table 2–8.

Table 2–8: Electrical Requirements

Parameter	DSRVF–BA	DSRVF–BB
Voltage Range	100-120 Vac (3-wire, single phase)	220-240 Vac
Frequency	50 to 60 Hz	50 to 60 Hz
Line Current	1.0 A	0.5 A
Power	75 W	75 W

2.3.4 Leakage Current

The DECserver 300 hardware leakage current is shown in Table 2–9.

Table 2–9: Leakage Current

Parameter	DSRVF–BA	DSRVF–BB
Voltage Range	100-120 Vac (3-wire, single phase)	220-240 Vac
Frequency	50 to 60 Hz	50 to 60 Hz
Leakage Current	0.68 mA	0.86 mA

2.4 Cables and Adapters

The following sections describe the cables and adapters which are used to connect the server to devices and to an Ethernet LAN

2.4.1 Ethernet Cables

The server can be connected to either a standard Ethernet or a ThinWire Ethernet LAN. Section 2.4.1.1 describes the cables used to connect the server to a standard Ethernet LAN and section 2.4.1.2 describes the cables used to connect the server to a ThinWire Ethernet LAN. Table 2–10 shows the maximum distance for different cable types.

Table 2–10: Maximum Cabling Distances

From	To	Maximum Distance	Cable Type
Server	Transceiver	50 meters (164 feet)	BNE3x–xx standard transceiver cable
Server	Transceiver	12.5 meters (41 feet)	BNE4x–xx office transceiver cable
Server	Power outlet	1.8 meters (6 feet)	Server power cable
Server †	DESPR/ DEMPR	185 meters (606 feet)	H8243–A cable

† No other device in ThinWire segment

2.4.1.1 Standard Ethernet Cables

The following transceiver cables are used to connect the server to a standard Ethernet LAN.

- BNE3x–xx IEE 802.3 standard transceiver cables
- BNE4x–xx IEE 802.3 office transceiver cables

BNE3x–xx standard transceiver cable is available in FEP versions, for use in return air conduits, and in PVC versions, for use in nonenvironmental airspaces. The large diameter of this cable results in a lower signal cable loss per length of cable than the smaller diameter office transceiver cable. Two styles of connectors are available: a straight connector and a right angle connector.

BNE3x–xx standard transceiver cables are available in lengths of 5 meters (16.4 feet), 10 meters (32.8 feet), 20 meters (65.6 feet), and 40 meters (131.2 feet).

BNE4x–xx office transceiver cable is available in PVC versions for use in non–environmental airspaces. The smaller diameter of this cable makes it ideal for use in office environments, however the smaller diameter results in a cable signal loss that is four times greater than that of BNE3x–xx transceiver cables. Two styles of connectors are available: a straight connector and a right angle connector.

BNE4x–xx office transceiver cables are available in lengths of 2 meters (6.6 feet) and 5 meters (16.4 feet).

2.4.1.2 ThinWire Ethernet Cables

ThinWire coaxial cables are used to connect the server to a ThinWire Ethernet LAN. This cable is available in FEP versions, for use in return air conduits, and in PVC versions, for use in nonenvironmental airspaces. Table 2–11 lists order codes and cable lengths for bulk ThinWire cables.

Table 2–11: ThinWire Coaxial Cable

Order Code	Type	Length
H8243-A	PVC	304.8 meters (1000 feet) reel
H8244-A	FEP	304.8 meters (1000 feet) reel

2.4.2 Device Port Cables and Adapters

The cables shown in Table 2–13 are used to connect to the device ports and Table 2–12 shows the maximum cabling distance. Table 2–14 describes the adapters which are used.

Table 2–12: Maximum Cabling Distances — Server to Devices

From	To	Line Speed	Maximum Distance	Cable Type
Server [1]	RS423/ EIA-423-A device	4.8 Kb/s	1200 meters (4000 feet)	H8245 or H8246 (24 AWG, 4 pair, twisted pair)
		9.6 Kb/s	900 meters (3000 feet)	
		19.2 Kb/s	300 meters (1000 feet)	
Server [1]	EIA-232-D device	4.8 Kb/s	75 meters (250 feet)[2]	24 AWG, twisted pair
		9.6 Kb/s	75 meters (250 feet)[2]	
		19.2 Kb/s	15 meters (50 feet)[2]	
Server [1]	EIA-232-D device	4.8 Kb/s	15 meters (50 feet)[2]	BC16E
		9.6 Kb/s	15 meters (50 feet)[2]	
		19.2 Kb/s	15 meters (50 feet)[2]	

[1] If it is necessary to use shielded cables, then the maximum distances must be de-rated because of increased cable capacitance. In such situations Digital recommends that 50 feet be considered the maximum.

[2] May be extended by using the H3105 active adapter .

Table 2–13: Device Port Cables

Order Code	Description
H8245 or H8246	4 pair, twisted pair, 1000 ft. reel. H8245-A is PVC version. H8246-A is plenum-grade.
BC16E-xx †	Terminated, 6-conductor cable
BC23P-10	10 ft, DECconnect SER cable, unshielded
BC23R-10	10 ft, DECconnect SER cable, shielded

† Available in 3.05 meter (10 feet), 7.62 meter (25 feet), and 15.24 meter (50 feet) lengths

Table 2–14: Device Port Adapters

Description	Order Code
MMJ loopback connector	H3103
25-pin passive adapter — female to MMJ	H8571-A
9-pin passive adapter — female to MMJ	H8571-B
25-pin passive adapter — male to MMJ	H8571-C
25-pin passive adapter — male to MMJ	H8571-D
25-pin passive adapter — male to MMJ	H8571-E
Cable concentrator (unshielded)	H3104
Cable concentrator (shielded)	H3125
Non DECconnect:	Part Number
Male to 50-way champ connector to eight MMPs	MOD-TAP 24-665-13
MMP to RJ12 (socket)	MOD-TAP 09-100-650

2.5 Port Devices Supported

The following sections list port devices that are supported by the server. For the latest listing of supported devices, see the *DECserver 300 Product Description* that applies to your operating system.

2.5.1 Terminals

The server supports Digital Equipment Corporation and non-Digital terminals that generate both 7-bit and 8-bit characters.

Digital terminals supported by DECserver 300 include:

- LA12, LA34, LA35, LA36, and LA38
- VT52
- VT101, VT102, VT125, VT131
- VT220, VT240, VT241
- VT320, VT330, VT340

2.5.2 Personal Computers

The server supports Digital and non-Digital personal computers in either terminal emulation mode or file transfer mode.

Digital personal computers supported by the server include:

- Professional 325, Professional 350, Professional 380
- Rainbow 100A, Rainbow 100B, Rainbow 100+
- DECmate II, and III
- VAXmate

Non-Digital computers supported by the server include:

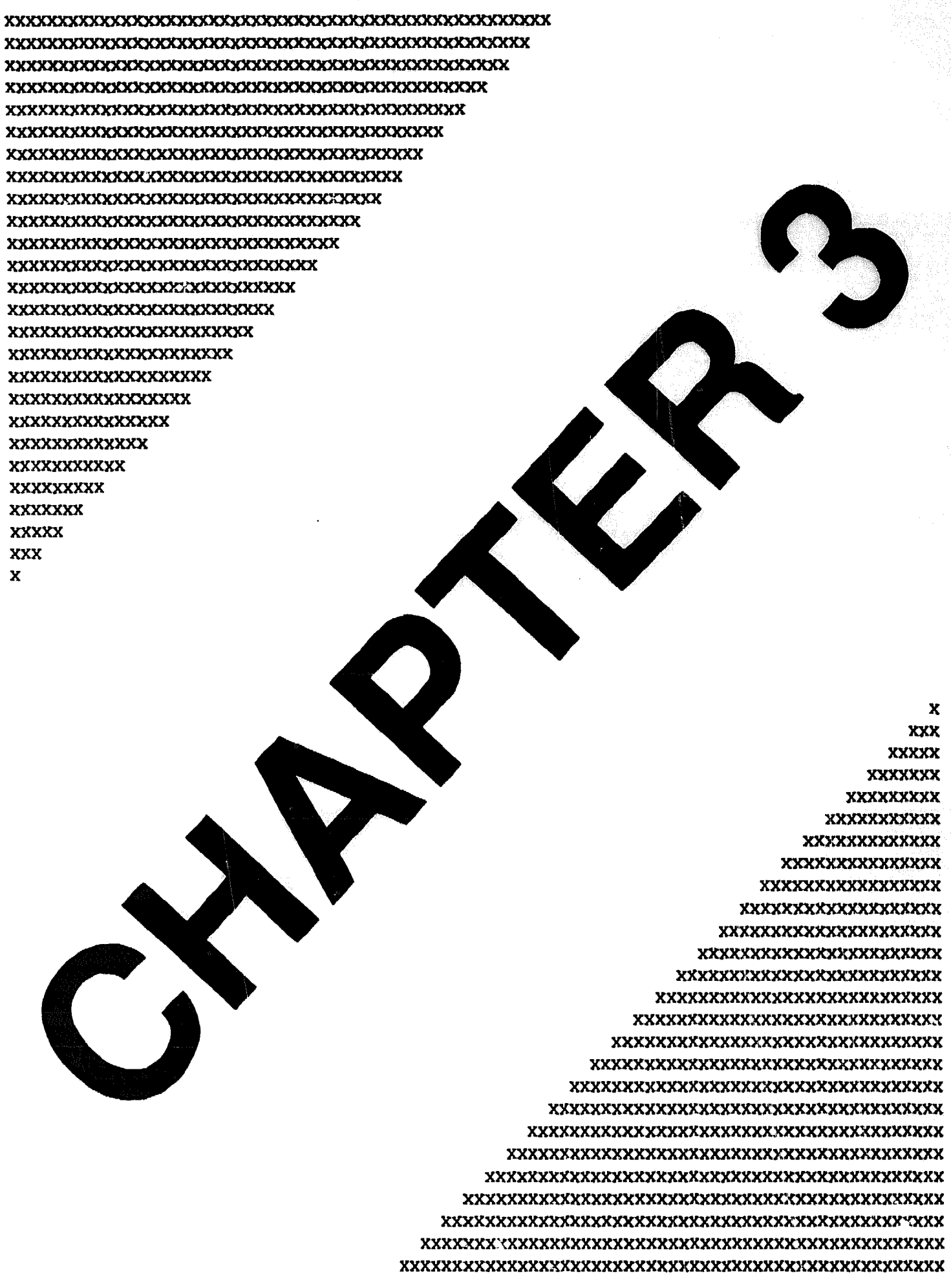
- IBM PC
- IBM PC/XT
- IBM PC AT

2.5.3 Printers

The server supports Digital and non-Digital printers that use RS-232-C (EIA-232-D) and EIA-423-A serial ports.

Digital printers supported by the server include:

- LA50, LA200, LA210
- LN01S, LN03S (Laser Printers)
- LCP01 (Ink Jet Printer)
- L1P02, LQP03 (Letter Quality Printers)
- LXY12-DA, LXY22-DA (Graphics Printers)
- LG01S, LG02 (Graphics Printers)
- DCT01, DCT03 (DECtalk)


CHAPTER 3

3

Server Hardware Functional Description

3.1 Introduction

The server has the following functional subsystems (see Figure 3–1):

- Central processing unit (CPU)
- Dynamic RAM (DRAM) and parity
- Peripheral
- Data path arbitrator (DPA)
- Network interface (NI)
- Initialization
- Refresh
- Power supply

The central processing unit (CPU) subsystem controls the server. It contains the 16.67 Mhz 68020 microprocessor, system clocks and their buffers, address decode and data signal acknowledge (DSACK) logic.

The dynamic RAM (DRAM) and parity subsystem contains 1 Mbyte of memory that is used by the 68020 microprocessor.

The peripheral subsystem, which is connected by the 16-bit B-BUS, controls the flow of data between the external port devices and the server. The peripheral subsystem also contains all of the internal circuitry that must interface with the 68020 microprocessor, except the DRAM and the network interface subsystem.

The peripheral subsystem contains the following circuits (see Figure 3–1):

- 16-bit B-BUS system
- Address decode logic
- Interrupt system
- Configuration register (Config. Reg.)
- Electrically erasable programmable read-only memory (EEPROM)
- Ethernet station address read-only memory (SAROM)
- Electrically programmable read-only memory (EPROM)
- Dual universal asynchronous receiver/transmitters (DUARTS)
- Serial Drivers/Receivers

The data path arbitrator (DPA) subsystem controls the sharing of memory between the 68020 microprocessor, the network subsystem, and the refresh logic.

The network interface subsystem controls the flow of data to and from the Ethernet. The network interface subsystem contains the following circuits:

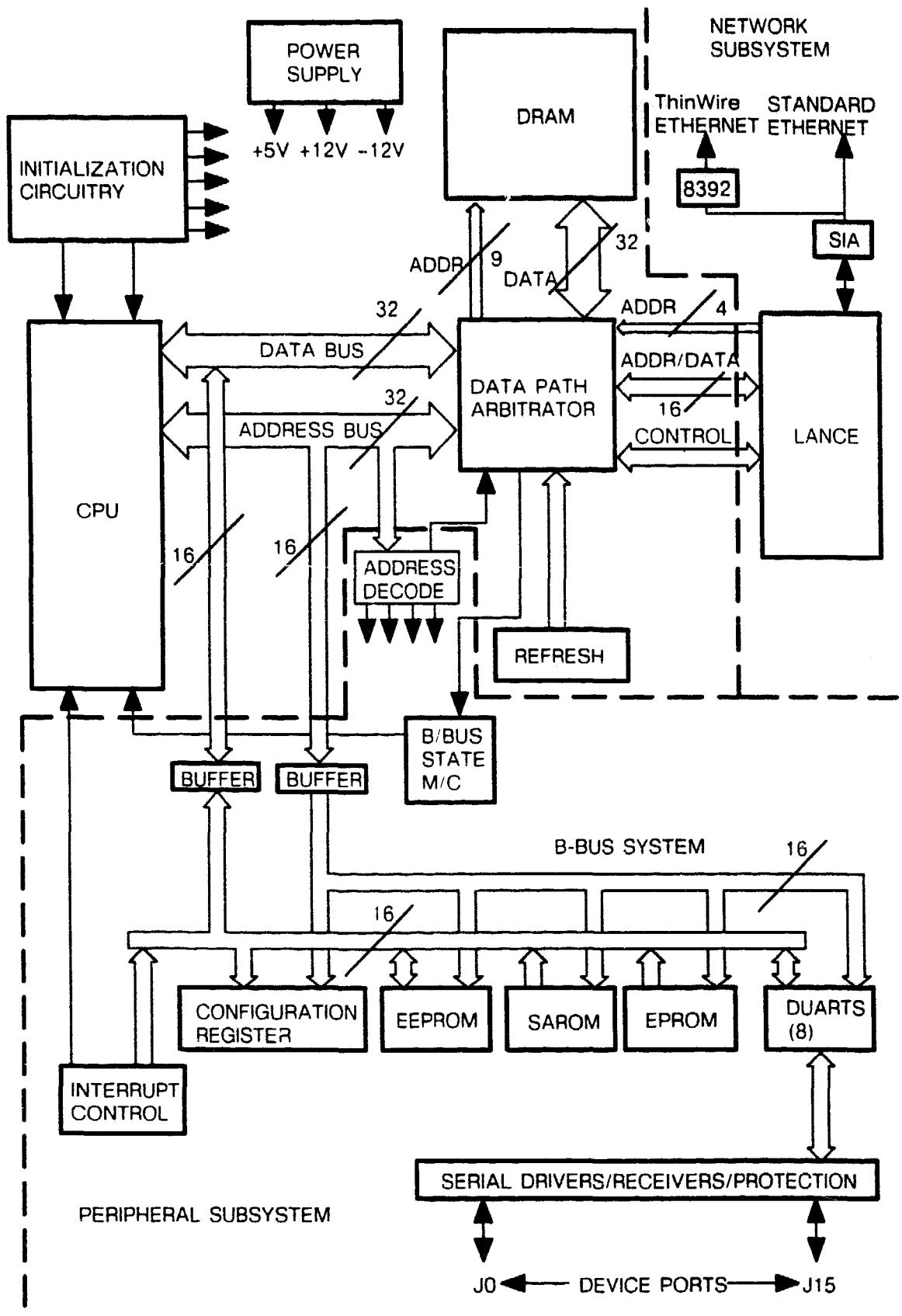
- Local area network controller for Ethernet (LANCE)
- Serial interface adapter (SIA)
- DP8392 transceiver

The initialization subsystem controls the initializing of all of the other subsystems and contains the reset timers.

The REFRESH subsystem refreshes the DRAM.

The power supply subsystem provides +5 Vdc, +12 Vdc, and -12 Vdc.

Figure 3-1: Server Hardware Overview



3.2 CPU Subsystem

The CPU subsystem consists of the following functional blocks:

- 68020 microprocessor
- System addressing
- Bus timer
- Clock generation logic

The following sections describe these functional blocks.

3.2.1 68020 Microprocessor

The 68020 microprocessor has a 32-bit data bus, a 32-bit address bus and is run at 16.67 Mhz. The microprocessor accesses the DRAM on a cycle-by-cycle basis as decided by the data path arbitrator (DPA). The DPA assigns priority as follows:

1. LANCE
2. REFRESH
3. 68020 microprocessor

The 68020 microprocessor communicates with the peripheral subsystem through a buffered 16-bit address bus and a buffered 16-bit data bus.

3.2.2 System Addressing

The 68020 microprocessor, with its 32-bit address bus, can address up to 4 gigabytes of memory. The server uses 1 megabyte of memory that is accessed by address lines A<19:0>. Address lines A<26:24> select which device is used. Address lines A<31:27> and address lines A<23:20> are not used. Table 3-1 shows the server memory address map.

Table 3-1: Server System Address Map

Devices	31	26	25	24	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EPROM	X[1]	0	0	0	X	X	X	X	A[2]	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
SAROM	X	0	0	1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	A	A	A	A	A	0
EEPROM	X	0	1	0	X	X	X	X	X	A	A	A	A	A	A	A	A	A	A	A	A	A	A	0
DUART0	X	0	1	1	X	X	X	X	X	X	X	X	X	X	X	X	0	0	0	A	A	A	A	0
DUART1	X	0	1	1	X	X	X	X	X	X	X	X	X	X	X	X	0	0	1	A	A	A	A	0
DUART2	X	0	1	1	X	X	X	X	X	X	X	X	X	X	X	X	0	1	0	A	A	A	A	0
DUART3	X	0	1	1	X	X	X	X	X	X	X	X	X	X	X	X	0	1	1	A	A	A	A	0
DUART4	X	0	1	1	X	X	X	X	X	X	X	X	X	X	X	X	1	0	0	A	A	A	A	0
DUART5	X	0	1	1	X	X	X	X	X	X	X	X	X	X	X	X	1	0	1	A	A	A	A	0
DUART6	X	0	1	1	X	X	X	X	X	X	X	X	X	X	X	X	1	1	0	A	A	A	A	0
DUART7	X	0	1	1	X	X	X	X	X	X	X	X	X	X	X	X	1	1	1	A	A	A	A	0
CON. REG.	X	1	0	1	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	A
LANCE[3]	0[4]	1	1	0	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	A	X
DRAM	X	1	1	1	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A

[1] X = Don't Care

[2] A = Address bit (1 or 0)

[3] As a slave.The 68020 can only perform word accesses to the LANCE when the LANCE is a slave. The LANCE, as master, can access DRAM only.

[4] The LANCE requires Address bit 31 to be 0. All other devices are X for bit 31.

The hexadecimal addresses that derive from Table 3-1 are shown in Table 3-2.

Table 3-2: Hexadecimal Addresses of Devices Accessed by the 68020 Microprocessor

Devices	Hexadecimal Addresses
EPROM	00000000 - 00FFFFFF
SAROM	01000000 - 01FFFFFF
EEPROM	02000000 - 02FFFFFF
DUART0	03000000 - 0300001E
DUART1	03000020 - 0300003E
DUART2	03000040 - 0300005E
DUART3	03000060 - 0300007E
DUART4	03000080 - 0300009E
DUART5	030000A0 - 030000BE
DUART6	030000C0 - 030000DE
DUART7	030000E0 - 030000FE
CON. REG.	05000000 - 05FFFFFF
LANCE	06000000 - 06FFFFFF
DRAM	07000000 - 07FFFFFF

3.2.3 Bus Timer

The bus timer ensures that the server will recover from incomplete bus cycles.

The timer detects errors in the reading and writing of data to and from the 68020, and expires. This causes the DPA, B-BUS and LANCE to be reset and a bus error exception routine to be run.

3.3 DRAM and Parity Memory Subsystem

The DRAM and parity memory subsystem is divided as follows:

- DRAM organization
- DRAM initialization
- DRAM addressing
- DRAM refresh

The following sections describe these functions in more detail.

3.3.1 DRAM Organization

The DRAM of the server appears, to the CPU and LANCE, to consist of 1 megabyte of dynamic RAM organized in a block 256K deep by 32 bits wide (see Figure 3–2). The data in DRAM is organized in longwords that are 4 bytes in width and a single parity bit is associated with each byte of data (see Figure 3–3). Each of the four data bytes can be accessed individually if required.

Figure 3–2: DRAM Organization

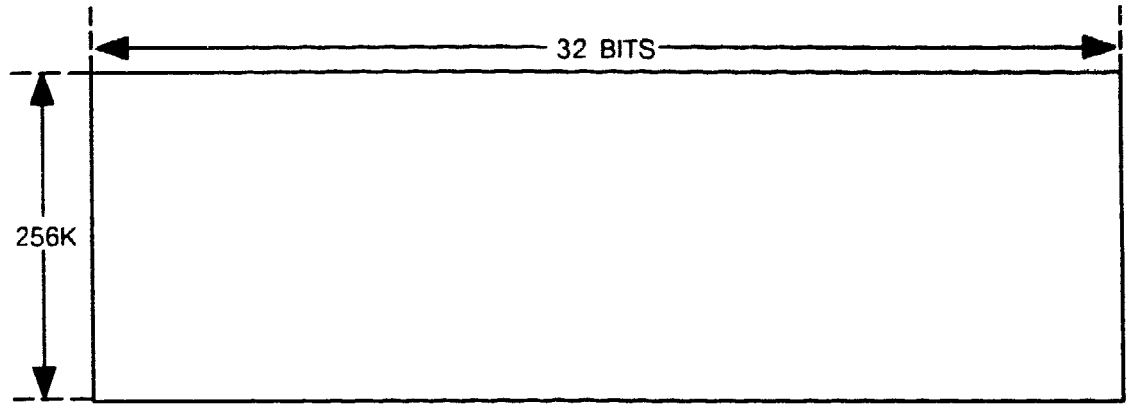
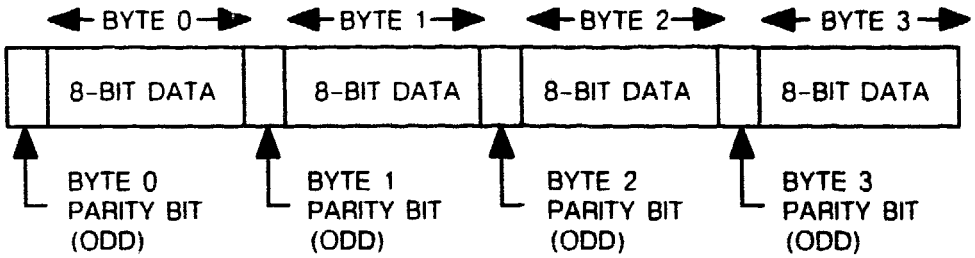


Figure 3–3: DRAM Organization — Data and Parity



Each time the CPU or LANCE writes to DRAM, they generate a parity bit per byte of data. The parity bit is written into parity DRAM at the same time the data is written into data DRAM. Only the parity bits of the bytes being written are effected during a write cycle.

Each time the CPU or LANCE reads from DRAM, all four bytes with their corresponding parity bits are read from DRAM and checked to see if a parity error exists. This is the case regardless of whether the data being read is byte, word, or longword in length. This means that before parity checking is enabled on power-up, data must be written to all locations in DRAM to avoid a false parity error being generated.

The Data Path Arbitrator (DPA) contains the parity generation and check logic.

3.3.2 DRAM Initialization

The DRAM takes about 200 microseconds to stabilize following power-up or hardware watchdog reset. A read operation of any DRAM location is carried out to initialize the internal refresh address counters. Following this read operation, data is written to all locations in DRAM to prevent a false parity error being generated (see Section 3.3.1). The DRAM is then ready for use.

3.3.3 DRAM Addressing

The DRAM has a 9-bit address bus and a data location is accessed by first putting the row address on the bus followed by the column address (RAS before CAS). The DPA decides what device (LANCE, CPU, or Refresh) accesses the DRAM and also multiplexes the required column and row address onto the 9-bit address bus.

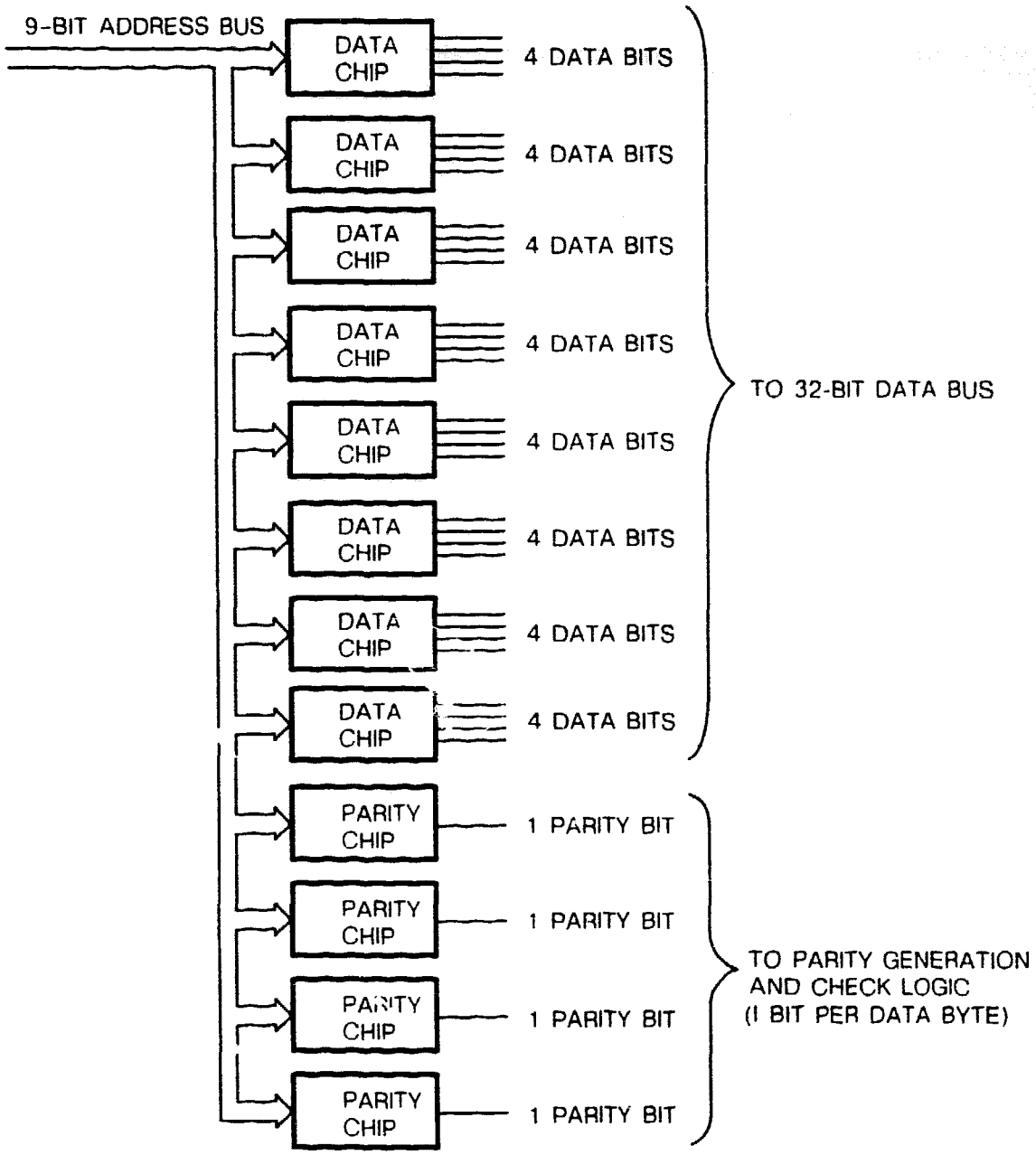
The DRAM memory is contained on 12 chips: 8 data DRAM chips and 4 parity DRAM chips (see Figure 3–4). The 9-bit address lines are connected in parallel to all 12 chips. Each data chip is divided into 512 rows by 512 columns and, when the row and column address has been decoded, outputs 4 data bits onto the data lines. This gives a total of 32 bits for the eight chips. Each parity chip is divided into 256k by 1 bit and, when the row and column address has been decoded, gives 1 parity bit per byte of data. This gives a total of 4 parity bits per longword address.

3.3.4 DRAM Refresh Subsystem

When Refresh accesses the DRAM, all columns corresponding to a particular row are refreshed simultaneously. Refresh uses the internal row address counter in DRAM to select the particular row to be refreshed.

The DRAM requires that all locations are refreshed every 8 ms. Therefore there must be 512 refresh cycles every 8 ms and the Refresh must prompt the DPA to gain access to the DRAM.

Figure 3-4: DRAM Addressing



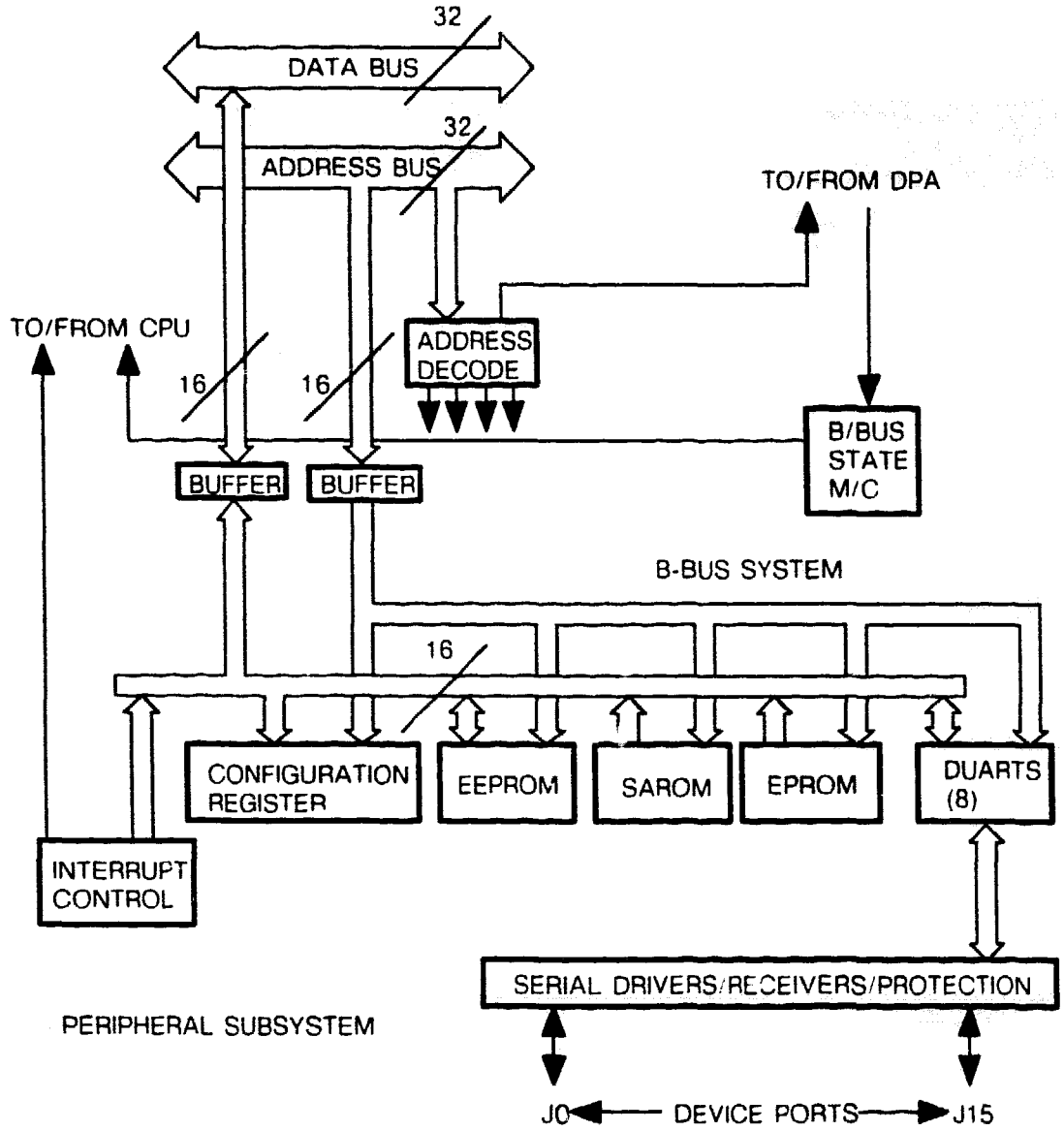
3.4 Peripheral Subsystem

The peripheral subsystem section of the server consists of the following (see Figure 3–5):

- B-BUS system
- Address decode logic
- Interrupt system
- Configuration register (Config. Reg.)
- Electrically erasable programmable read-only memory (EEPROM)
- Station address read-only memory (SAROM)
- Electrically programmable read-only memory (EPROM)
- Dual Asynchronous Universal Receiver/Transmitters (DUARTS)
- Serial drivers/receivers

The following sections describe these circuits.

Figure 3-5: Peripheral Subsystem



3.4.1 The B-BUS System

The B-BUS system is 16-bit buffered bus system that connects the peripheral devices to one another and to the upper 2 bytes of of the CPU data bus.

3.4.2 Address Decode Logic

The address decode logic selects and enables the relevant device on the B-BUS to respond to the address currently on the 16-bit address bus.

3.4.3 Interrupt System

When an interrupt is raised the interrupt system encodes the priority level of the interrupt and notifies the 68020 microprocessor. The 68020 recognizes the level of interrupt and requests the vector address from the B-BUS system. The B-BUS responds by sending the vector address or requesting that the 68020 microprocessor automatically generate a vector.

The 68020 microprocessor has seven levels of interrupts. Level 7 is the highest priority and level 1 the lowest. Table 3–3 shows how the interrupt priority levels are assigned.

Table 3–3: Interrupt Priority Level Assignments

Level	Interrupting Device
7	Parity error
6	Software watchdog timer
5	Receiver interrupt (DUART)
4	CTS interrupt †
3	LANCE interrupt
2	General timer interrupt
1	Transmit interrupt (DUART)

† Only used if DUARTS are configured for DSR/DTR use.

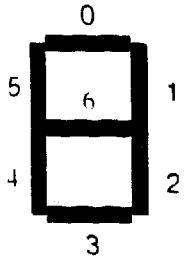
3.4.4 Configuration Register

The configuration register is a 16–bit wide, write only, register. The lower seven bits control the seven segment display. The most significant bit (MSB) of the lower byte indicates self-test is in progress, and the upper eight bits are used for diagnostics. Table 3–4 describes the bit definitions for the configuration register and Figure 3–6 shows the segment definition for the seven–segment display.

Table 3-4: Configuration Register Bit Definition

Bits	Name	Description
<15>	DUARTCLR H	When clear, resets the DUARTs
<14>	Loopback H	When set, causes RTS to be loopbacked to CTS on all DUARTs
<13>	N/C	
<12>	N/C	
<11>	N/C	
<10>	EEPROMEWE H	When set, enables writes to EEPROM (NVRAM)
<9>	HWCLR	When transitioned from 0 to 1 clears hardware watchdog timer
<8>	PERRCLR	When clear, disables and clears parity error interrupts
<7>	SELFTEST H	When set, indicates self-test is in progress
<6>		When set, turns off segment 6 When clear, turns on segment 6
<5>		When set, turns off segment 5 When clear, turns on segment 5
<4>		When set, turns off segment 4 When clear, turns on segment 4
<3>		When set, turns off segment 3 When clear, turns on segment 3
<2>		When set, turns off segment 2 When clear, turns on segment 2
<1>		When set, turns off segment 1 When clear, turns on segment 1
<0>		When set, turns off segment 0 When clear, turns on segment 0

Figure 3-6: Seven-Segment Display — Segment Definition



3.4.5 Electrically Erasable Programmable Read-Only Memory (EEPROM)

The EEPROM is also known as NVRAM (Non-Volatile RAM) and is used to store the server permanent database and fault history. The EEPROM has an 8 kilobyte capacity.

3.4.6 Station Address Read-Only Memory (SAROM)

The SAROM is a 32-byte Programmable read-only memory (PROM) that is used to store the Ethernet Address of the server.

3.4.7 Electrically Programmable Read-Only Memory (EPROM)

The EPROM has 64 kilobytes and and stores the firmware code. The firmware code includes self-test, maintenance operation protocol (MOP), and utilities. EPROM also contains the factory settings for the server's parameters.

3.4.8 Dual Universal Asynchronous Receiver/Transmitters (DUARTS)

The eight DUARTS convert the parallel data from the B-BUS to serial data for transmission to the serial drivers, and convert the serial data from the serial receivers to parallel data for the B-BUS.

Each DUART consists of two serial full-duplex asynchronous receiver/transmitters, a 16-bit programmable counter/timer, a 7-bit input port and an 8-bit output port.

Both the hardware watchdog timer and the software watchdog timer use DUART timers. The following sections describe the the function of these timers.

3.4.8.1 Hardware Watchdog Timer

The hardware watchdog timer ensures that the server can recover from a halted processor or corrupted code that the software watchdog timer cannot handle.

The timer is programmable and is loaded during initialization and reloaded regularly by the software. If the software crashes, it is unable to reload the hardware watchdog timer and the timer expires. This causes a hardware watchdog timer to set a flip-flop and to reset the entire module with the exception of the lower byte of the configuration register and the flip-flop. Bit 7 of the configuration register is the self-test bit.

When the firmware recognizes the hardware watchdog reset signal it does the following:

- Dumps the contents of DRAM to the host

- Starts self-test
- If self-test is successful, loads software-image from host and transfers control to software

3.4.8.2 Software Watchdog Timer

The software watchdog timer ensures that the server can recover if the software goes into a loop or stops.

The timer is programmable and is loaded during initialization and reloaded regularly by the software. When the software stops or goes into a loop, the timer expires and causes a level 6 interrupt. The level 6 interrupt calls a software routine that, in turn, calls the firmware. The firmware goes through the same routine as for a hardware watchdog timer timeout.

3.4.9 Serial Drivers/Receivers

The serial drivers/receivers convert outgoing data/signals from the DUARTS to EIA-423-A format for transmission via the 16 asynchronous ports. The serial drivers/receivers also convert incoming data/signals from the 16 asynchronous ports to a suitable format for input to the DUARTS. EIA-423-A is compatible with the EIA-232-D interface, but supports longer cabling runs and higher cabling speeds. In addition, EIA-423-A minimizes damage to computing equipment caused by electrical overstress (EOS) and electrostatic discharge (ESD). EOS and ESD are the most common cause of communications equipment failure.

3.5 Data Path Arbitrator(DPA)

The DPA controls access to DRAM by

- CPU
- LANCE
- REFRESH

The DPA provides an interface for these devices, acknowledges requests for use of the DRAM and places a priority on each request. When the DPA grants access to DRAM, by CPU or LANCE, it multiplexes the row and column address onto the DRAM address lines and checks for parity error (read cycle) or generates parity bit (write cycle).

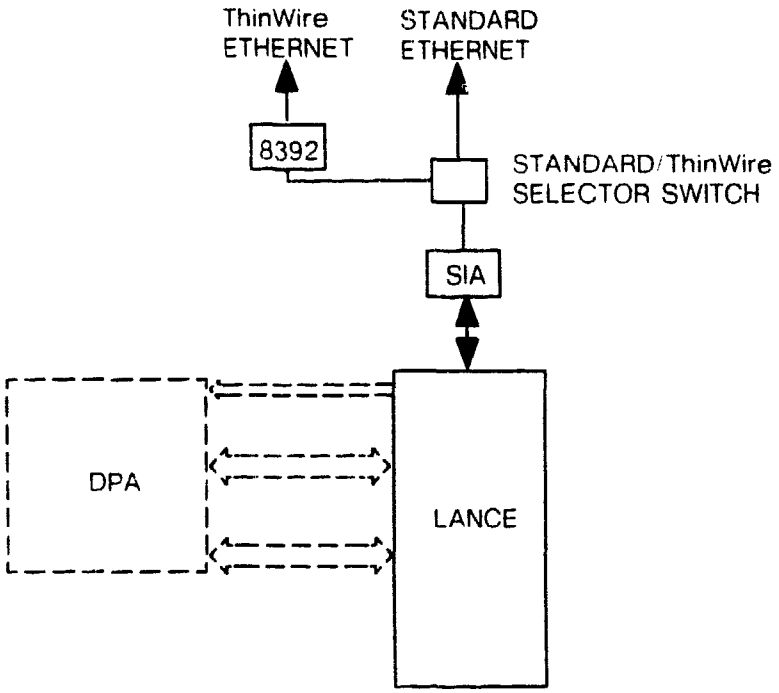
3.6 Network Interface Subsystem

The network interface subsystem consists of three functional blocks (see Figure 3-7):

- DP8392 coaxial transceiver interface (CTI)
- Serial interface adapter (SIA)
- Local area network controller for ethernet (LANCE)
- Standard/ThinWire selector switch

These functional blocks are described in greater detail in the following sections.

Figure 3–7: Network Interface Subsystem



3.6.1 DP8392 Coaxial Transceiver Interface (CTI)

The DP88392 coaxial transceiver interface (CTI) connects to the ThinWire Ethernet via a BNC connector. The CTI performs transmit, receive, and collision detection functions for the network controller.

3.6.2 Serial Interface Adapter (SIA)

The SIA performs manchester encoding and decoding of the data transferred over the network. Also, the SIA filters noise and interprets collisions for the LANCE circuit.

3.6.3 Local Area Network Controller (LANCE)

The LANCE converts data between the network serial format and the system byte-wide format. The LANCE operates in two modes:

- Slave mode — The LANCE accepts instructions and control information from the DPA.
- Master mode — The LANCE transmits and receives data to and from DRAM using Direct Memory Access (DMA).

The LANCE functions in master mode, when receiving data,as follows:

1. Receives information from SIA
2. Converts the serial network bit stream into a parallel, 8-bit wide, stream.
3. Strips the Ethernet preamble and synchronization pattern.
4. Checks and removes the CRC bits
5. Uses DMA to place the data in DRAM.

The LANCE functions as follows, in master mode, when transmitting data:

1. Uses DMA to read data from system memory.
2. Converts the data to a serial bit stream.
3. Adds a preamble and sync pattern.
4. Calculates and adds the CRC at the end of the data packet
5. Passes the data packet to the SIA for transmission on the Ethernet

3.6.4 Standard/ThinWire Selector Switch

The standard/ThinWire selector switch chooses between standard and ThinWire Ethernet.

3.7 Initialization Subsystem

The initialization subsystem controls the resetting and initialization of the the other subsystems in the server. The initialization subsystem consists of the following functional blocks:

- Power-up reset logic
- Hardware watchdog timer logic
- Bus timer reset logic
- Server database reset switch

3.7.1 Power-Up Reset

The power-up reset signal is asserted when power is applied to the server. The power-up reset signal resets the following components:

- 68020 microprocessor
- LANCE
- DPA
- B-BUS
- Refresh Circuit
- Configuration register, causing the following:
  - Turns on the seven-segment display
  - Turns on the diagnostic LED
  - Resets the DUARTs
  - Disables parity checking
  - Enables hardware watchdog reset circuit
- Bus timer circuit
- Hardware watchdog timer

NOTE

The 68020 reset instruction has no effect on external logic.

3.7.2 Hardware Watchdog Timer Reset

The hardware watchdog timer reset logic enables the server to recover from corrupted software code or a halted processor. The hardware watchdog timer reset logic resets the following components:

- 68020 microprocessor
- LANCE
- DPA
- B-BUS
- Configuration Register, upper byte only, causing the following:
  - Turns on the diagnostic LED
  - Resets the the DUARTs

- Disables parity checking
- Enables hardware watchdog reset circuit

The hardware watchdog timer reset logic does *not* reset the following:

- Refresh circuit
- Lower byte of the configuration register
- Hardware watchdog timer
- Bus timer

3.7.3 Bus Timer Reset

The bus timer reset circuit enables the server to recover from incomplete bus cycles. The bus timer reset logic resets the following components and then resets itself:

- LANCE
- B-BUS
- DPA

The bus timer reset circuit does *not* reset the following:

- 68020 microprocessor
- Refresh circuit
- Configuration register
- Hardware watchdog timer
- Bus timer

3.7.4 Server Database Reset Switch

On power-up, if the server database reset switch is pressed for a minimum of five seconds, the firmware loads the factory settings into the permanent database in EEPROM (NVRAM) from EPROM.

3.8 Module Jumpers (For Manufacturing and Repair Use Only)

The server provides extra utilities for fault finding during repair and manufacturing which are enabled by placing jumpers on J4 on the main board (see Figure 3–8). The most commonly used are:

- Self-Test — Manufacturing Mode

Enabled by connecting pin 1 to pin 9. This causes the self-test to run in manufacturing mode (see 5.2.2 for further information)

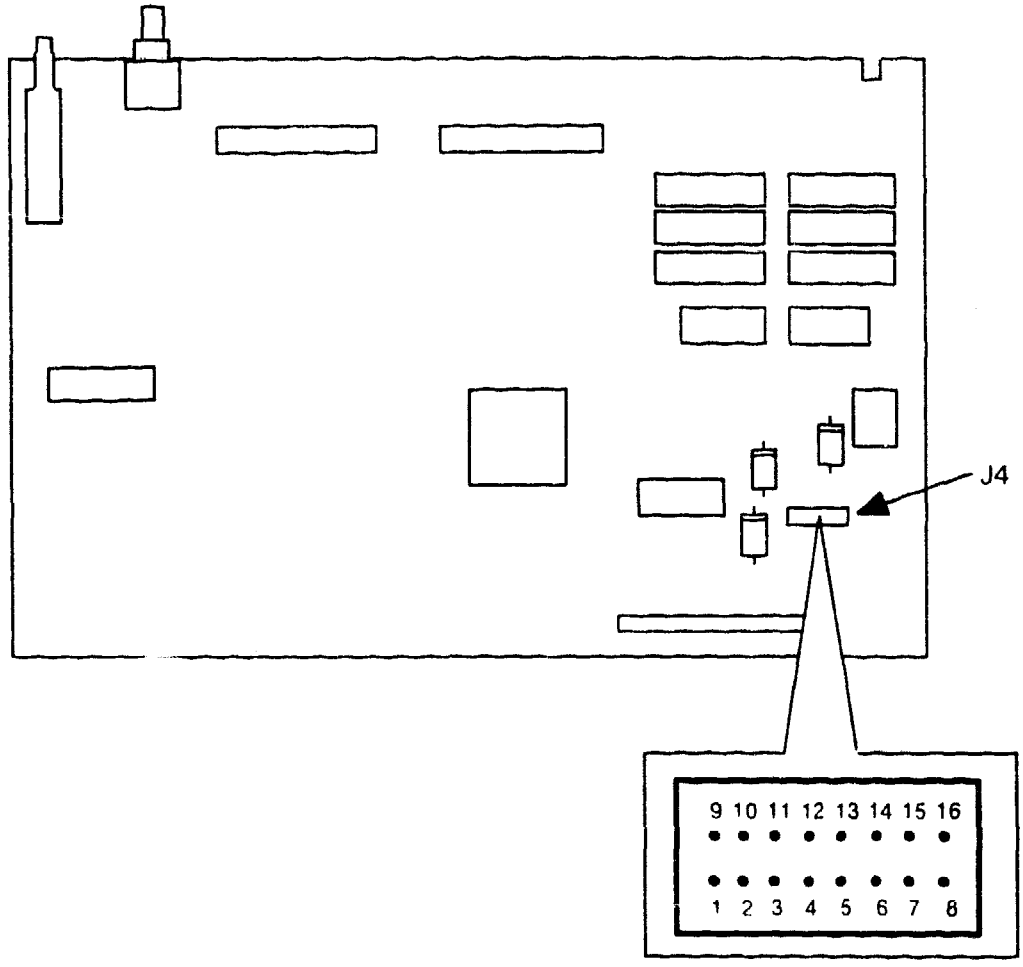
- Console Command Interface (CCI)

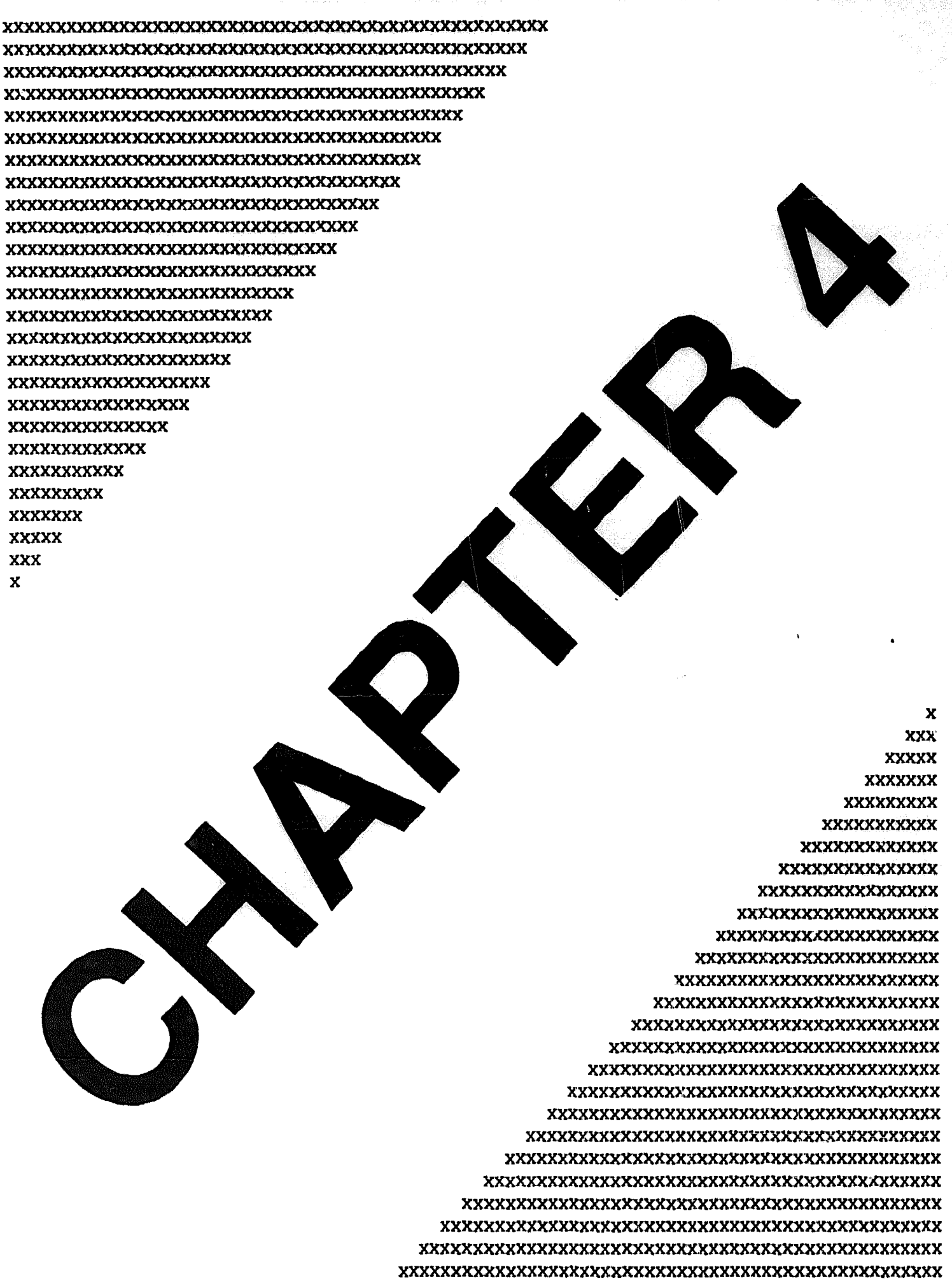
  Enabled by connecting pin 4 to pin 12. When enabled, CCI runs self-test in console mode (see 5.2.3 for further information) .

- Trigger Pin

  Pin 14 of J4 outputs a trigger pulse when a fatal error occurs.

Figure 3-8: Location of J4 on the Main Board




CHAPTER 4

4

Software Functional Operation

4.1 Introduction

This chapter provides an overview of server software operation. The functional components of the software are described in a basic manner. The information is provided as a supplement to the information provided in the *Local Area Transport (LAT) Network Concepts* manual.

4.2 Server Software Components

The server software is divided into the following functional components:

- User interface handler
- Maintenance operation protocol (MOP) handler
- Local Area transport protocol (LAT) handler
- Ethernet driver
- Port driver
- TD/SMP (terminal device/session management protocol) handler
- Initialization routines
- General purpose routines
- On-Line help

4.2.1 User Interface Handler

The user interface handler consists of two sub-components that analyze and process commands received from the port driver:

- Parser
- Action routines

Among the inputs handled by the user interface handler are the following:

- CONNECT, DISCONNECT, RESUME and LOGOUT Commands
- On-line HELP
- LIST, MONITOR, and SHOW Commands
- TEST Commands

4.2.2 Maintenance Operation Protocol (MOP) Handler

MOP routines exist both in the software and in the firmware. The information in this section applies to the routines in the software.

The MOP handler is responsible for the following functions:

- Remote console functions:
  - Providing server system-ID when requested
  - Periodic broadcasting of server system-ID
  - Ethernet data link counters
  - Remote console carrier links
  - Boot
- Loopback

The NCP CONNECT command and software utilities such as Terminal Server Manager use remote console carrier links to manage the server from another node.

The loopback facility allows a message to be looped via the server to the point of origin.

4.2.3 Local Area Transport (LAT) Protocol Handler

The LAT protocol handler implements both the master and slave sides of the the LAT protocol for the server.

The LAT protocol handler is responsible for the following functions:

- Virtual circuit processing
- Slot processing
- Service class functions:
  - Offering services
  - Queuing connection requests
  - Maintaining service and service-node directories
  - Multicast message transmission and reception

  CONNECT, DISCONNECT, and RESUME command processing

For further information on LAT protocol, see the *Local Area Transport (LAT) Network Concepts* manual.

4.2.4 Ethernet Driver

The Ethernet driver controls the operation of the LANCE as follows:

- Handles interrupts generated by LANCE
- Handles requests from other server software components to use Ethernet
- Maintains the Ethernet data link counters and enables access to these counters

4.2.5 Port Driver

The port driver controls the operation of the sixteen asynchronous ports as follows:

- Handles interrupts
  - Places data from DUARTs into specific port input buffers
  - Controls selected physical port flow control protocol
  - Invokes user interface handler or LAT protocol handler
- Notifies user interface handler or LAT protocol handler when a transmit operation to a port device is complete
- Handles requests from other server software components to transmit data through the ports
- Maintains the counters for each port as well as the current status

4.2.6 TD/SMP Handler

The TD/SMP handler manages multiple sessions on devices attached to the ports.

4.2.7 Initialization Routine

The initialization routine is called by the firmware when the server image has been down-line loaded correctly and does the following:

- Copies permanent database from NVRAM into DRAM. The permanent database contains:
  - Server characteristics
  - Port characteristics
  - Service characteristics
- Initializes all of the DUARTS
- Initializes LANCE
- Initializes hardware and software watchdog timers

4.2.8 General Purpose Routines

The general purpose routines component has the following functions:

- Memory management
- Timer services for process scheduling
- Communications between various software functions
- General purpose library routines that are commonly used by all software components

4.2.9 On-Line Help

The on-line help component provides interactive help to the operator.

4.3 Sample Data Flow

As an example of how some of the server software modules work together, the data flow sequence involved in processing a CONNECT BIGVAX server command is described in the following paragraphs. Before the command is entered at the user's terminal, the following assumptions are made about the state of the server and the service node offering the service BIGVAX:

- The user interface handler has transmitted a local prompt (Local>) character string out to the terminal by means of the port driver.
- The user interface handler is now awaiting an input notification from the port driver for the port that is servicing the terminal.
- The service BIGVAX is available.
- There are no existing sessions on the server to BIGVAX.

4.3.1 Establishing the Virtual Circuit

To establish a virtual circuit, the data flow is as follows:

- As the user enters characters in the CONNECT command, the port driver services the resulting interrupts.
- The port driver then places the characters into the command input buffer for the port in question and invokes the user interface handler.
- The user interface handler then scans the newly received data for a carriage return and processes any special editing characters received such as DELETE or Control/R.
- When the user interface handler detects the carriage return, it invokes the ASCII parser.
- Because the CONNECT command is valid, the ASCII parser action routines execute. The action routines determine the service node with the best rating for the service BIGVAX by scanning the node database in server memory. Then the routines save the required node and service information in a block of memory that contains an area for the port servicing the terminal.
- The user interface handler then calls the LAT protocol handler and passes on two pieces of information:
  - The location of the port's connect command area in server memory
  - Notification that a new session is to be established
- The LAT protocol handler scans the virtual circuit database in server memory to determine whether a virtual circuit has already been established to the selected service node.
- Finding that no virtual circuit exists, the LAT protocol handler then calls the memory management routines to allocate enough memory pool for the data structures required to maintain the new session.
- If there is enough memory for the virtual circuit, the LAT protocol handler calls the memory management routines once again to allocate enough memory for the data structures required to maintain the new session.

- If not enough memory is available for either the circuit or the session, the LAT protocol handler informs the user interface handler of the inability to service the request, and the user interface handler requests the port driver to output the appropriate error message to the user's terminal.

- If there is enough memory available to establish the circuit and the session, the LAT protocol handler attempts to create a virtual circuit to the desired service node by formatting a LAT Start Circuit message in a previously allocated Ethernet transmit buffer and then by requesting that the Ethernet driver give the transmit packet to the LANCE for transmission.

- When the service node responds by transmitting a LAT Start Circuit message back to the server, the LANCE executes a Direct Memory Access (DMA) operation to copy the received messages into one of the receive buffers assigned to it by the Ethernet driver, and raises an interrupt.

- The server CPU then invokes the interrupt handler in the Ethernet driver, and the Ethernet driver determines which buffer contains the message just received.

- The Ethernet driver them examines the message to determine the protocol type and informs the LAT protocol handler of the new buffer, the buffer's physical memory location, and other information associated with the buffer.

- The LAT protocol handler determines that the virtual circuit is now established and exits to wait for the virtual circuit timer to expire. The circuit timer must expire before the LAT protocol handler can transmit any further messages to this virtual circuit.

4.3.2 Establishing the Session

To establish a session, the data flow is as follows:

- When the virtual circuit timer expires, the timer routines inform the LAT protocol handler of that event. The LAT protocol handler then locates the area in memory that describes the virtual circuit and and the location of the areas for the session(s) on that circuit. In our example, there is only one session area needed.

- The LAT protocol handler then formats a LAT run circuit message with an enclosed Start Slot in an Ethernet transmit buffer. A LAT start slot is essentially a request to start a session, and the slot is constructed from information found in the sessions area that the server maintains. The transmit buffer containing the run circuit message is then passed to the Ethernet driver which in turn gives the buffer to the LANCE for transmission.

- When the service BIGVAX responds with its own run circuit message having an enclosed start slot, the LAT protocol handler is invoked as before by means of the Ethernet driver.

- The LAT protocol handler uses information in the received LAT message header to locate the virtual circuit area in server memory for this circuit. The LAT protocol handler then looks further into the message for the slot header that will help the server locate the session area in server memory. The slot header indicates that a start slot has been received thus showing that a session has been established.

4.3.3 Logging In and Exchanging Data

With the session established, the user can now use the terminal to communicate with service node BIGVAX. Typically, a VMS service node will invoke a log-in process after the LAT session has been established. The service node log-in process sends the user name prompt (Username:) to the user at the terminal. The data flow for the exchange of data is as follows:

- The user name prompt is delivered to the server by a LAT run circuit message from the service node that contains a LAT data slot header, followed by the prompt string that constitutes the data.

- The LAT protocol handler then requests the port driver to start a transmit operation to the port connected to the user's terminal starting at the address of the first byte of data in the received data slot. The user then sees the username prompt appear on the terminal.

- As the user types in characters in response to the user name prompt, the port driver places the characters into a server session input buffer assigned for that LAT session.

- When the next virtual circuit timer expires, the timer routines invoke the LAT protocol handler. The LAT protocol handler checks to see if any sessions on its virtual circuits have data to be transmitted to the service nodes at the other end of the virtual circuits. If a virtual circuit has pending data, the LAT protocol handler builds a LAT run circuit message to send to the service node and includes a data slot for the user who just typed some characters in response to the user name prompt.

- The LAT protocol handler then requests the Ethernet driver to give the run circuit message to the LANCE for transmission to the service node.

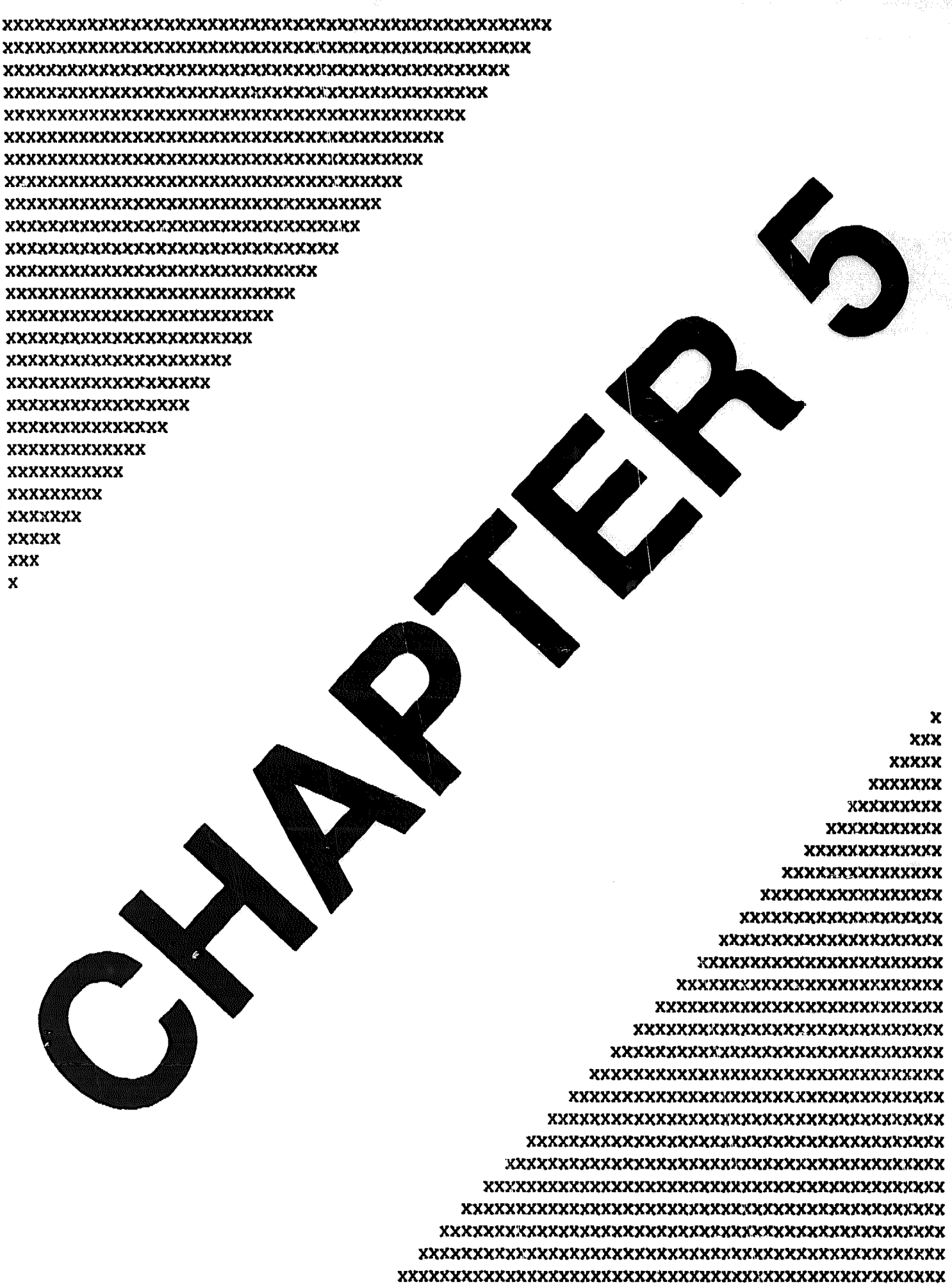
- This entire process continues until the session is disconnected by either the user or the service node.

4.3.4 Other Interactive Processes

The data flow described in section 4.3.3 is a simplified discussion. A large number of interactive processes were left out. These processes include:

- Error handling in the virtual circuit
- Slot operations other than those used for starting a session and sending user data
- Software module interchanges that occur when a virtual circuit goes into an idle state
- Managing multiple sessions on a single circuit

Virtual circuits go into an idle state when there is no data available to transmit over the virtual circuit from either the server or the service node.


CHAPTER 5

5

Firmware Functional Operation

5.1 Introduction

The firmware is divided into three major components:

- Self-test
- Maintenance operation protocol (MOP)
- Repair/debug utilities

The following sections describe these components in greater detail.

5.2 Self-Test

Self-test has four modes:

- Normal mode
- Manufacturing mode
- Console mode
- Server entry mode

5.2.1 Normal Mode

In normal mode, entered on power-up, self-test executes each test once and then transfers control to MOP if no fatal error is detected.

If a non-fatal error is detected, error information is written to NVRAM, the diagnostic dot is set blinking, and self-test proceeds to the next test.

NOTE

Firmware documentation and reference material use the generic terms non-volatile random access memory (NVRAM), random access memory (RAM), and read-only memory (ROM). The type of NVRAM used in the server is electrically erasable programmable read-only memory (EEPROM). The type of RAM used is dynamic RAM (DRAM) and the type of ROM used is electrically programmable read-only memory (EPROM).

If a fatal error is detected, error information is written to non-volatile memory, the corresponding error code will be flashed on the seven-segment display, and self-test will halt.

5.2.2 Manufacturing Mode

Manufacturing mode is entered on power-up when the manufacturing mode jumper is installed. Manufacturing mode will continuously execute self-test. This allows self-test to run continuously while in a burn-in chamber, or at a repair station and is used during server repair or manufacture.

5.2.3 Console Mode

Console mode provides different ways to run self-test and is used during server repair or manufacture.

5.2.4 Server Entry Mode

Server entry mode is invoked by the server software command INITIALIZE. In Server entry mode the server can:

- Loop self-test a specified number of times
- Disable external loopback on the Ethernet port
- Select extended RAM test

Further information on the INITIALIZE command can found in the *Terminal Server Commands and Messages Reference* manual.

5.2.5 Self-Test Routines

The self-test routines are as follows:

- Bootstrap test
- RAM subsystem test

- Interrupt subsystem test
- Timers reset test
- ROM subsystem test
- Network interface subsystem test
- DUART subsystem test
- System-wide test

The following sections describe the tests in greater detail.

5.2.5.1 Bootstrap Test

The bootstrap test checks the 68020 address and data registers, and the portion of DRAM that is used by the firmware.

5.2.5.2 RAM Subsystem Test

The RAM subsystem test consists of the following tests

- RAM refresh logic test

  This test verifies that the the DRAM Refresh circuit is operational.

- RAM transfer size test

  This test verifies that the 68020 and the data path arbitrator cooperate to size the DRAM data bus correctly. Longword, word, and byte operations to RAM are tested.

- 68020 vector base register test

  This test verifies that the 68020 vector base register (VBR) functions correctly.

- 68020 cache test

  This test verifies that the processor executes instructions faster when the cache is enabled.

- RAM stuck-at fault test

  This verifies that the DRAM is free from stuck-at faults

- RAM coupling fault test (disabled in normal mode)

  This test verifies that no coupling faults exist within the DRAM chips.

- RAM primary address test

  This test verifies that the address decoding logic (both external and internal to the DRAM chips) allows each DRAM location to be independently addressed.

- RAM parity test

  This test verifies that a parity interrupt is generated when a DRAM location containing bad parity is read. It also verifies that parity DRAM is free from stuck-at faults.

5.2.5.3 Interrupt Subsystem Test

The interrupt subsystem test consists of one test:

- Spurious interrupt test

  This test verifies that no spurious interrupts, caused by stuck-at interrupt lines or faulty peripherals, occur when the processor is dropped to zero.

5.2.5.4 Timers Test

The timers test consists of the following tests:

- Watchdog timers test

  This test verifies the hardware watchdog timer and the software watchdog timer.

- General purpose timer test

  This test verifies the general purpose timer.

5.2.5.5 ROM Subsystem Test

The ROM subsystem test consists of the following tests:

- EPROM CRC test

  This test verifies the integrity of the EPROM by calculating the cyclic redundancy check (CRC) value of its contents and comparing it with the value stored in the last location of EPROM.

- NI address test

  This test calculates the checksums of the NI address stored in the NI address ROM, using the method outlined in the Ethernet Specification V2.0, and compares it with the checksum stored in the NI address ROM. In addition, the test checks the known pattern also stored in the ROM.

- EEPROM read/write test

This test verifies that the EEPROM may be read and written to correctly.

- EEPROM checksum test

  This test computes the checksum of each of the parameter block in EEPROM and compares it with the value in the EEPROM. A checksum failure is not fatal. If the comparison fails then the default values for the failing block, obtained from firmware EPROM, are used.

5.2.5.6 NI Subsystem Test

The NI subsystem test consists of the following tests:

- LANCE register test

  This tests verifies that the LANCE CSRs are free from stuck-at faults.

- Accept physical address test

  This test verifies that the LANCE will accept an internal loopback packet addressed to it.

- Reject physical address test

  This test verifies that the LANCE will reject an internal loopback packet that is not addressed to it.

- Transmit CRC logic test

  This test verifies that the LANCE can correctly generate a good CRC and append it to a packet

- Receive CRC logic test

  This test verifies that the LANCE can correctly receive a packet with a good CRC.

- Receive bad CRC test

  This tests verifies that the LANCE will flag an error when it receives a packet with a bad CRC.

- Collision detect logic test

  This tests verifies the LANCE collision detect logic by using a test mode of the LANCE that causes it to generate a collision each time it attempts to transmit.

- Accept broadcast package test

  This test verifies that the LANCE will correctly recognize the broadcast address of all ones.

- Accept/reject multicast packet test

This test verifies the ability of the LANCE to accept and reject multicast addresses.

- BYTESWAP mode test

  This test verifies that the LANCE can swap packet bytes when BYTESWAP mode is enabled.

- External loopback test

  This test verifies the continuity of the NI port data path. It requires that the selected NI port have a loopback connector installed.

5.2.5.7 DUART Subsystem Test

The DUART subsystem test consists of the following tests:

- Receive interrupt, character length, and parity test

  This test verifies that each channel can generate a receive interrupt, and generate and check the various character sizes and parity configurations.

- Break generate/detect test

  This test verifies that each channel can generate and detect a break condition.

- Detect framing error test (disabled in normal mode)

  This tests verifies that each channel can detect framing errors by running transmit and receive at different speeds.

- Detect overrun error test

  This test verifies that overrun error is correctly reported when the receive FIFO overflows.

- Transmit/receive at different baud rates test

  This tests verifies that each channel can transmit and receive at various baud rates. In this test all channels are active simultaneously.

- System-wide exerciser test

  The purpose of this test is to generate as much activity as possible, in order to detect interaction errors not visible when each subsytem is tested individually. The LANCE, DUART, timers, and interrupt subsystems are exercised simultaneously.

5.2.6 Error Reporting

The self-test reports errors as follows:

- Seven-segment display

- Console port (non-fatal errors)

The seven-segment display is located on the rear of the server (see Figure 5–1). The decimal point of the seven-segment display is known as the diagnostic dot and also provides diagnostic information (see Table 5–1). The seven-segment error codes are described in Table 5–2.

Figure 5–1: Seven-Segment Display

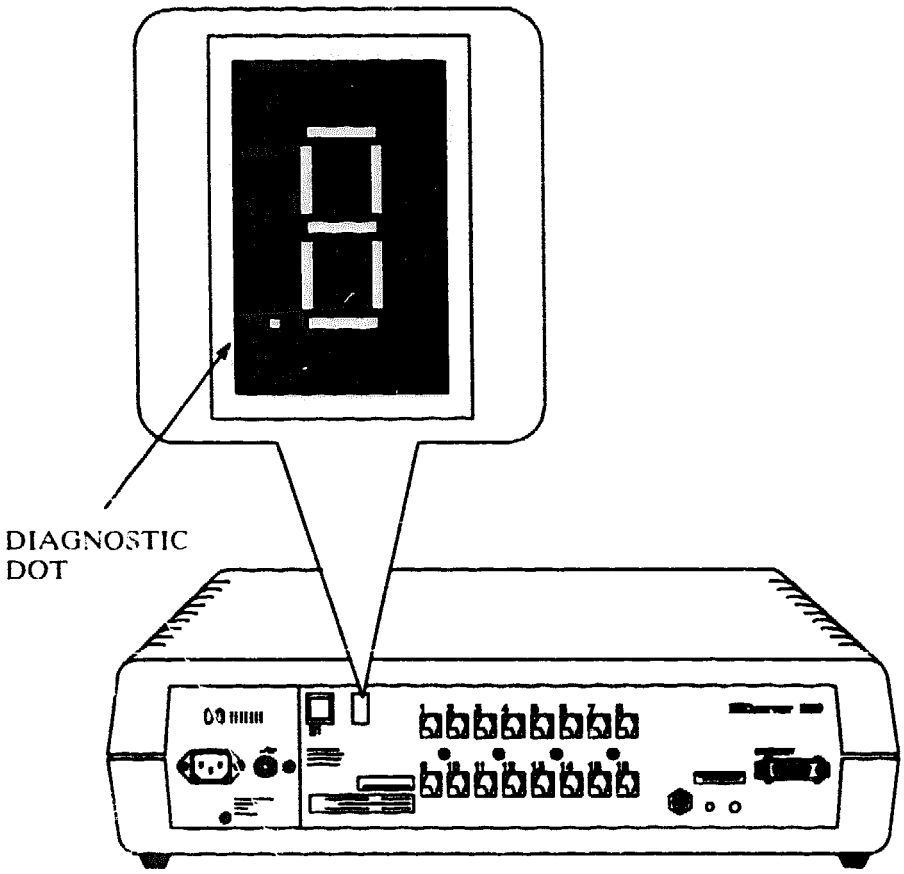


Table 5–1: Diagnostic Dot Display

Diagnostic Dot	System Status
On	No fatal errors
Off	Fatal error or self-test in progress
Blinking	Non-fatal error detected

Table 5-2: Seven-Segment Display - Error Codes

Display	System Status
F	Bootstrap tests executing
E	RAM subsystem test executing
d	Interrupt subsystem tests executing
C	Timer tests executing
b	ROM subsystem tests executing
A	Ethernet subsystem tests executing (internal loopback)
9	Ethernet subsystem external loopback test executing
7	Asynch subsystem tests executing (internal loopback)
6	Asynch subsystem tests executing (external loopback) †
5	System exerciser tests executing

† Not executed in normal mode

The self-test outputs non-fatal error information to the designated console port in the server database (The default is port 1). In order to access this information, check that a terminal is connected to the console port and that the port parameters are set for a baud rate of 9,600 and a character size of 8 bits with no parity. The appropriate error messages can then be seen on the terminal as show in Table 5-3.

Table 5-3: Console Port Terminal Error Display

Console Terminal	Diagnostic Dot	Seven-Segment Display
Local -920- Parameter checksum error on port(s) n	Blinking	Rotating Segment
Local -921- Factory-set parameters will be applied to port(s) n	Blinking	Rotating Segment
Local -922- Port hardware error on port(s) n	Blinking	Rotating Segment
Local -923- Port n will be disabled	Blinking	Rotating Segment
Local -926- DSR/DTR signal error on port(s) n	Blinking	Rotating Segment
Local -925- Port(s) may be used with data leads only	Blinking	Rotating Segment
Local -930- Server parameters checksum error	Blinking	Rotating Segment
Local -931- Factory-set parameters will be applied	Blinking	Rotating Segment
Local -935- Service characteristic checksum error	Blinking	Rotating Segment
Local -936- Service will be disabled	Blinking	Rotating Segment
Local -941- Transceiver Loopback Error †	Blinking	9
Local -942- Image Load not attempted	Blinking	9
Local -950- Troubleshooting procedures should be followed	Blinking	9
Local -952- Enter ^P to restart selftest	Blinking	9

† This is the only fatal error where the diagnostic dot blinks and the seven-segment display does not flash the display character.

5.2.7 Running Self-Test

Self-test executes when you:

1. Power-up the server
2. Execute the INITIALIZE command
3. Execute a LOAD or TRIGGER command on a DECnet host
4. Press CTRL/P on the console terminal if an error prevented down-line load during previous self test

The diagnostic dot is turned off as self-test starts and the seven-segment display counts down from "F" through "5" as each block of tests is executed. If a fatal error is detected, testing stops and the seven-segment display flashes the code corresponding to the failing test. If a non-fatal error is detected, testing continues and the diagnostic dot blinks.

NOTE

There is one exception to this sequence. If the server fails the Ethernet subsystem external loopback test, the display stops at "9" and the diagnostic dot blinks. The "9" on the display does not blink.

When the server passes self-test, control is handed over to MOP and MOP attempts to find a host and load the software image.

NOTE

When power is applied to the server, the seven-segment display shows an "8" and the diagnostic dot is on for approximately 0.5 second as a display check.

5.3 Maintenance Operation Protocol (MOP)

MOP provides a number of functions in firmware:

- Down-line load
- Up-line dump
- Remote console
- Loop

The seven-segment display supplies information on the status of the server during loading, dumping and software execution (see Table 5–4).

Table 5–4: Seven-Segment Display Status Codes

Display	System Status
4	Requesting load
3	Load request backoff
2	Loading
1	Requesting dump
0	Dumping
(Rotating segment)†	Server software executing

† Segment rotates, outlining a "figure-eight" pattern

5.3.1 Down-Line Load

When the server passes self-test, control is handed to MOP to down-line load the server image. The down-line load process has three phases as shown below:

- Load request

  The server transmits load requests in an attempt to locate a load host. The seven-segment display shows a "4" during load request.

- Load backoff

  The server refrains from transmitting load requests for a period of time when previous requests have not resulted in a successful load. This is to reduce the burden on the available load hosts. The backoff period starts at 4 seconds and doubles each time, up to a maximum of 5 minutes. The seven-segment display shows a "3" during load backoff.

- Image transfer

  Loading of image after load host is located. The seven-segment display shows a "2" during image transfer.

The backoff state in only entered if the server has been unable to load a software image. Normally, the server goes from the "load request" state to the "image transfer" state.When the software image has been loaded, MOP transfers control so that the software image can initialize the server.

The DECserver 300 down-line load process supports both Ethernet and ISO8802-2/3 (IEEE 802.2/802.3) on the LAN. The "Load Request" state is comprised of two parts, one of which uses ISO8802 format, and the other Ethernet format. When the server makes a load request, it first attempts to load using ISO8802 format. If unsuccessful, the server then attempts to load using Ethernet format. If the server is unsuccessful in both formats it enters the backoff state. At the end of the backoff state the server enters the load request state and starts again.

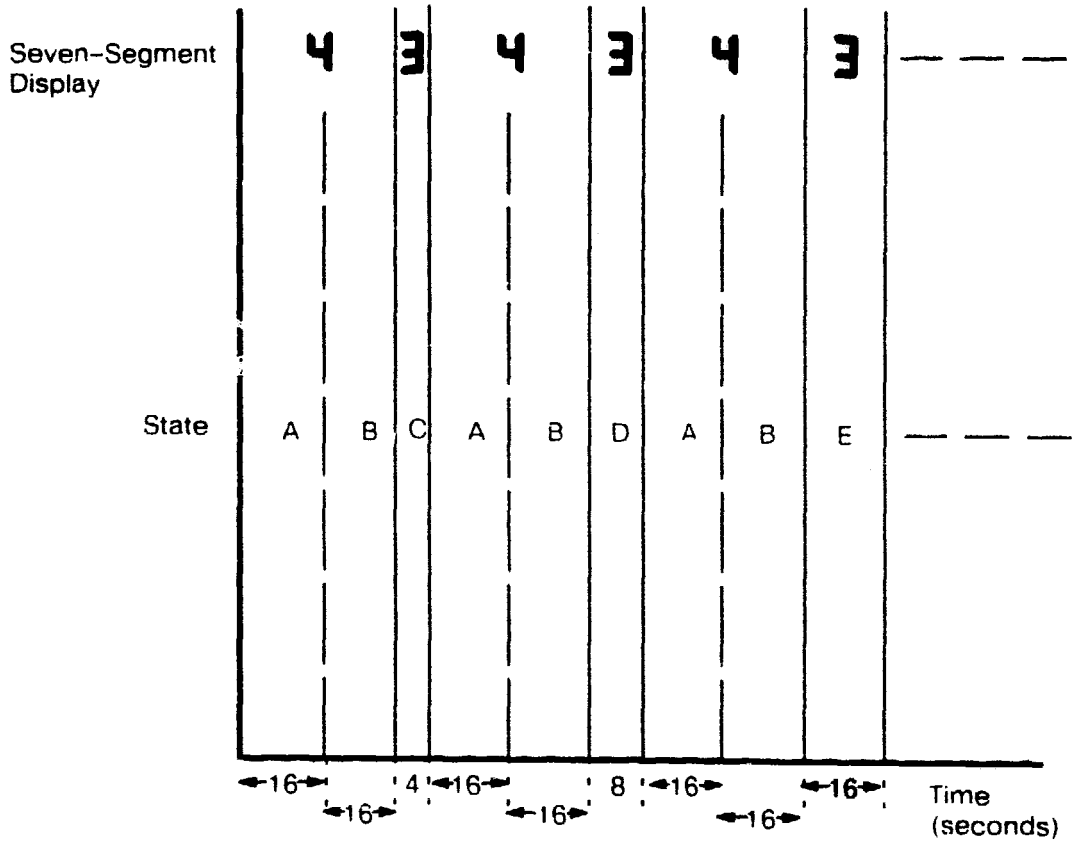
Figure 5–2 shows the loading process when the server is unable to locate a load host. The diagram shows the use of the two datalink formats, the changes between the load request and backoff states as a function of time, and the corresponding seven-segment display codes.

As well as seven-segment display (see Table 5–4) the console port terminal supplies information on the status of the down-line load (see Table 5–5).

Table 5–5: Console Port Terminal – Status Display

Console Terminal	Symptoms	Seven-Segment Display
Local -912- Load failure, timeout	Down-line load starts, then fails.	3
Local -916- Illegal load image, load aborted	Down-line load starts, then fails.	3
Local -953- Attempting to locate load host, [ISO8802] Local -953- Attempting to locate load host, [ETHERNET]	Down-line load does not start	4
Local -951- DECserver will retry operation in *n* seconds	Down-line load does not start	4

Figure 5-2: Down-Line Load - Server Unable to Locate Host



A = Load Request, ISO8802 format
B = Load Request, Ethernet format
C = Backoff (1st, 4 seconds)
D = Backoff (2nd, 8 seconds)
E = Backoff (3rd, 16 seconds)

4 = Load Request

3 = Backoff

5.3.2 Up-Line Dump

The software invokes up-line dump when it detects a logic error or when the command CRASH is used. The up-line dump function stores the contents of DRAM on a host for subsequent examination.

5.3.3 Remote Console

The remote console program provides the following functions:

- Providing server system-id when requested
- Periodic broadcasting of server system-ID
- Ethernet data link counters
- Remote console carrier links
- Boot

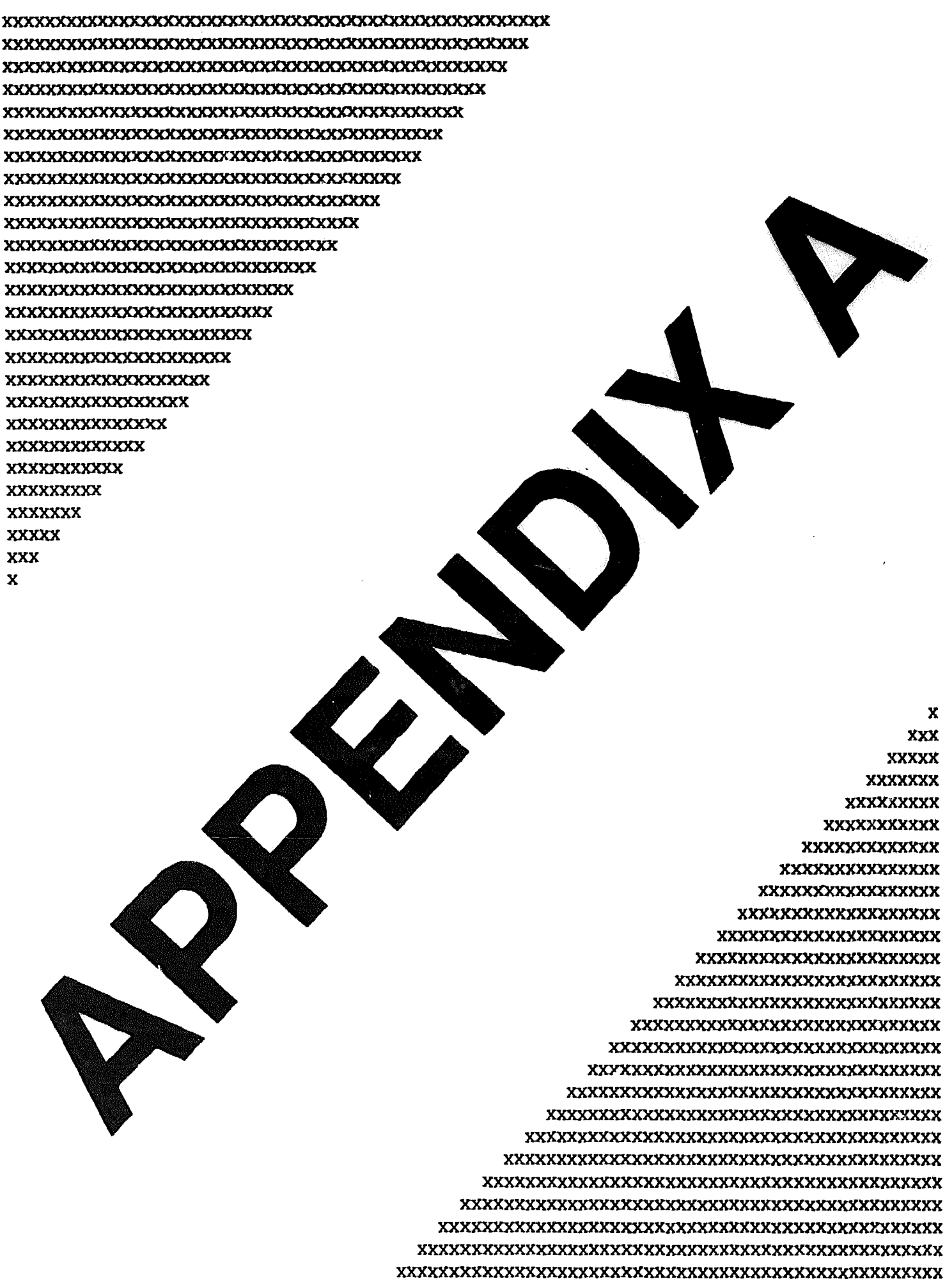
5.3.4 Loop

The loop program enables the server to loop back packets over the Ethernet.

5.4 Repair/Debug Utilities

The server supplies the following utilities for to aid developers and repair and manufacturing personnel.

- Console command interface (CCI)
- On-line debugging tool (ODT)
- Self-test — manufacturing mode


APPENDIX A

A

MOP System ID Format for the DECserver 300 Server

This appendix provides basic information to help you decode the maintenance operation protocol (MOP) System ID messages that are generated by the server. The MOP protocol and message formats used by the DECserver 300 server are described in the *DNA Maintenance Operations Functional Specificaton V3.0* (Order No. AA-X436A-TK). Use this appendix in conjunction with that specification.

As a Digital Ethernet node, a DECserver 300 terminal server identifies itself on the Ethernet by transmitting a system identification message every 8 to 10 minutes. The server also responds on demand to system identification requests from other nodes on the Ethernet by transmitting this system identification back to the requesting node as soon as possible.

These system identification messages are often used by the server's load hosts for down-line loading, for Ethernet-level tests, and for establishing remote console facility (RCF) sessions.

User-written software applications running on Ethernet nodes can also read MOP system ID messages to extract information about terminal servers that are active on the network. Complete instructions on how to write software to intercept these messages is beyond the scope of this manual. However, this appendix and the MOP specification provide enough information for you to decode these messages.

The MOP system ID message transmitted by the server contains several separate blocks of information about the server. The kind of information found in each block is identified by the INFO TYPE field. Each INFO TYPE field is followed by a 1-byte INFO LENGTH field, followed by the INFO VALUE field. The INFO VALUE field is a set of bytes that contain the actual data for the block.

The INFO TYPEs in every MOP system ID message generated by an Ethernet node are as follows (see the MOP specification for the required encoding of the INFO VALUE fields):

INFO TYPE Value	Information	Description
1	MAINTENANCE VERSION	MOP protocol version used by server
2	FUNCTIONS	MOP functions supported by this node
3	HARDWARE ADDRESS	Server's physical Ethernet address
4	COMMUNICATION DEVICE	INFO VALUE = 60 for DECserver 300

Since the DECserver 300 server supports RCF, the server is also required to include these additional INFO TYPEs:

INFO TYPE Value	Information	Description
3	CONSOLE USER	Ethernet address of current RCF user
4	RESERVATIONS TIMER	RCF session timeout threshold in seconds
5	CONSOLE COMMAND SIZE	Server RCF command buffer size in bytes
6	CONSOLE RESPONSE SIZE	Server RCF response buffer size in bytes

When the DECserver 300 firmware is in control of the DECserver 300 hardware, a DECserver 300 MOP system ID message includes all the INFO TYPEs mentioned thus far. The firmware controls the server when one of the following occurs:

- Self-test is in progress
- A down-line load is in progress
- An up-line dump is in progress

After a down-line load of the DECserver 300 software image, the server software includes all the INFO TYPEs mentioned above in every MOP system ID message. In addition, it can include these two optional INTO TYPEs.

INFO TYPE Value	Information	Description
400	DATA LINK TYPE	INFO VALUE = 1 (Ethernet)
401	DATA LINK BUFFER SIZE	INFO VALUE = 1518

MOP designates INFO TYPEs 101–199 for Ethernet nodes to communicate additional information in their MOP system ID messages. Within this range, the server uses INFO TYPEs 102–106 to broadcast the following device-specific information about itself.

INFO TYPE Value	Information	Description
101	DIAGNOSTIC STATUS	Server status after last self-test
102	ROM VERSION (3) : B =	Version, ECO version, sub-ECO version
103	SOFTWARE VERSION (3) : B =	Version, update version, test version
104	SERVER NUMBER (2) : B =	Two-byte binary server number
105	SERVER NAME (1-16) : A =	ASCII string, size in INFO LENGTH field
106	SERVER ID (1-17) : A =	ASCII string, size in INFO LENGTH field

The data for INFO TYPEs 101-106 appears in the DECserver 300 SHOW SERVER displays in Section 9.5 of the *DECserver 300 Management* manual.


INDEX

Index

Numbers

A

B

C

D

E

F

H

I

J

L

M

N

O

P

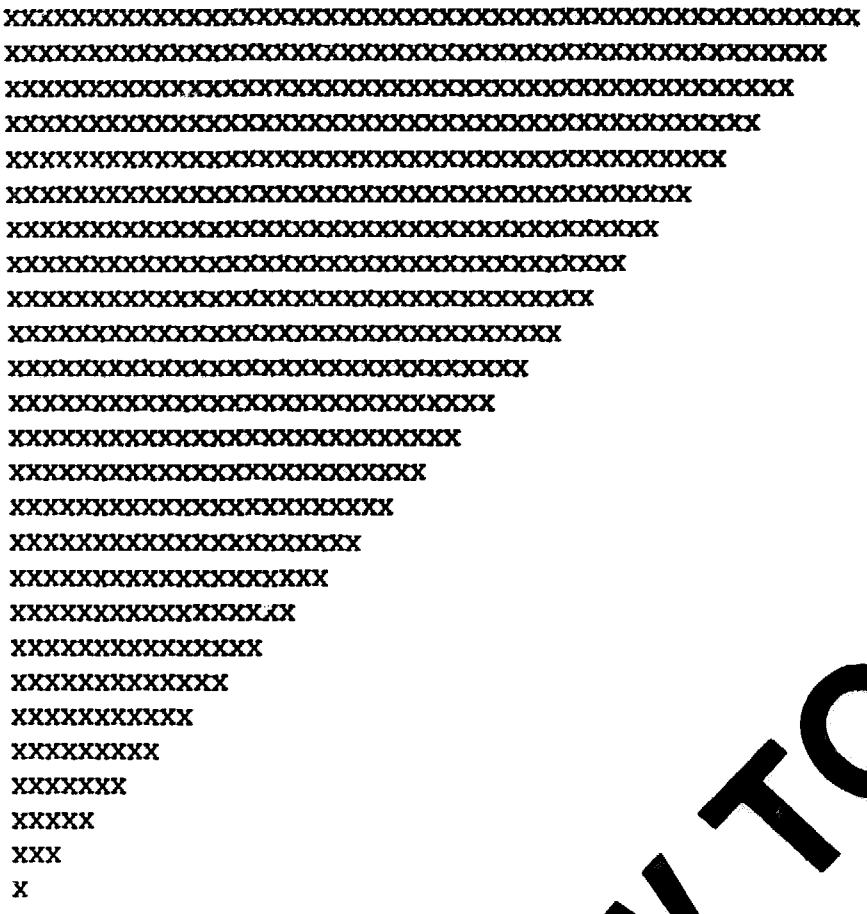
R

S

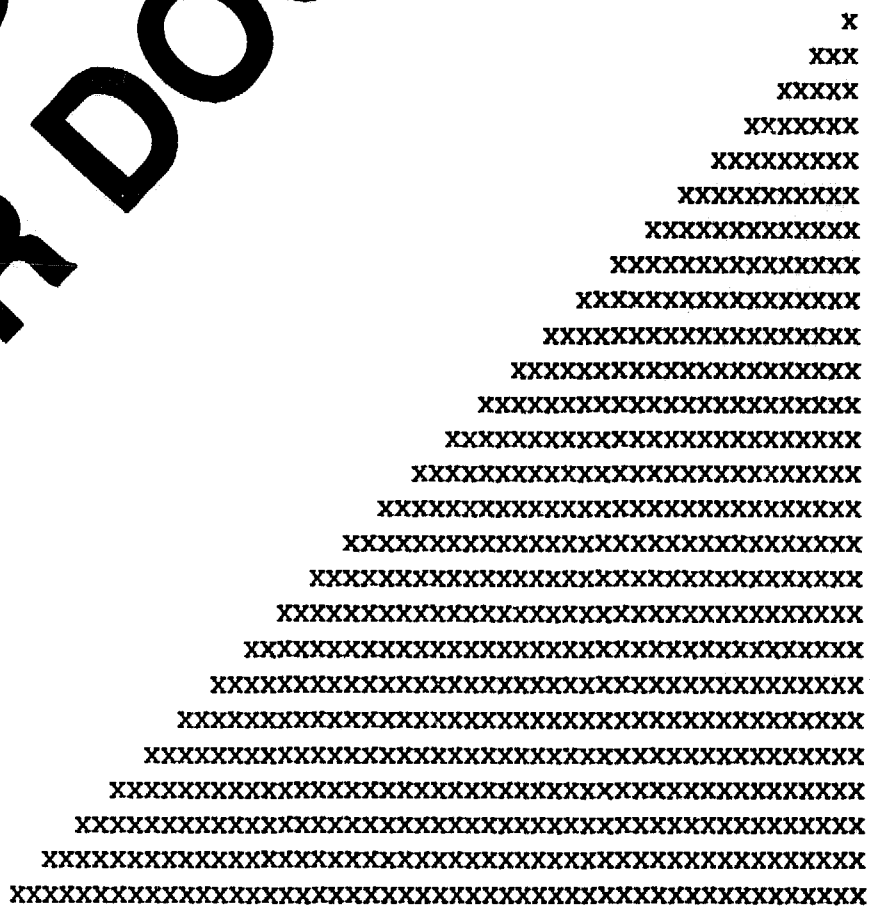
T

U

V



HOW TO ORDER DOCUMENTS



How to Order Documents

All the information you need to order additional documents is provided here. The ordering procedure depends on:

- Whether you are a customer or a Digital employee.
- Your location: USA or Puerto Rico, Canada, or other.
- Your means of placing the order: telephone, electronic mail, or regular mail.

The specific ordering procedures and the order numbers for software and hardware manuals are provided in the following sections.

Order Numbers

For software manuals, use the documentation kit order number. For hardware manuals, use the document order number.

VMS Software Documentation Kits

The DECserver 300 documentation kit order number is QA-VTUAA-GZ. The kit contains one of each of the following manuals:

- *DECserver 300 Introduction*
- *Local Area Transport (LAT) Network Concepts*
- *DECserver 300 Management*
- *Terminal Server Commands and Messages*
- *DECserver 300 Problem Solving*
- *DECserver 300 Use*
- *DECserver 300 Commands Quick Reference*

- *Terminals Server User's Reference Card*
- *DECserver 300 Software Installation*
- *Terminal Server Glossary*

TSM Documentation Kit

The TSM documentation kit order number is QLZ42-GZ. This kit is intended for the installer and manager of the TSM software product and contains two manuals:

- *Guide to Terminal Server Manager*
- *Terminal Server Manager Software Installation Guide*

ULTRIX-32 Software Documentation kit

The DECserver 300 ULTRIX-32 documentation kit order number is QA-VTVAA-GZ. This kit contains the same manuals as the VMS documentation kit except that the ULTRIX-32 kit contains the *DECserver 300 Software Installation (ULTRIX-32)* instead of the VMS version.

User Software Documentation Kit

The user documentation kit order number is QA-VTUAB-GZ. This kit is intended for the terminal users and contains only two manuals:

- *DECserver 300 Use*
- *Terminal Server User's Reference Card*

Hardware Documents

The order numbers for the hardware manuals are listed below.

Hardware Documents

Manual Title	Order Number
DECserver 300 Hardware Installation	EK-A0366-IN
DECserver 300 Identification Card	EK-A0368-IC
DECserver 300 Technical Description	EK-A0367-TM
DECconnect System Installation and Verification Guide	EK-DECSY-VG
DECconnet System Planning and Configuration Guide	EK-DECSY-CG
Support Print Set (Digital Internal Use Only)	EM-02270-01

Ordering Procedures for Customers

If you are a customer, refer to the tables below for ordering information.

How to Order by Phone

If You Live In	Call
USA	(800) DIGITAL
Puerto Rico	(800) 754-7575 x2012
Canada	(800) 267-6215

How to Order by Mail

If You Live in	Write to
USA or Puerto Rico[1]	Digital Equipment Corporation, P.O. Box CS2008, Nashua, New Hampshire 03061
Canada	Digital Equipment of Canada LTD., 940 Belfast Road, Ottawa, Ontario, Canada KIG 4C2 ATTN: A&SG Business Manager
Other	Digital Equipment Corporation, A&SG Business Manager, c/o Digital's local subsidiary or approved distributor

[1]Any prepaid order from Puerto Rico must be placed with the Local Digital Subsidiary: (809) 754-7575 x2012

Ordering Procedures for Digital Employees

Software Documentation Kits

Use the *Internal Software Order Form* to order software documentation kits. The form is available from office services and the Software Service Bureau (SSB).

Hardware Manuals

You can place your order for hardware manuals by telephone, VAXmail, or DECmail as follows:

- Telephone number: (617) 351–4323 (DTN: 234–4322)
- DECmail address: ORDER @NRO