Prime ®

System Administrator's Guide,
Volume II: Communication
Lines and Controllers

Revision 21.0

DOC10132-1LA

SYSTEM ADMINISTRATOR'S GUIDE, VOLUME II: COMMUNICATION LINES AND CONTROLLERS

First Edition

Lois Anne Conrad

This guide documents the software operation of the Prime Computer and its supporting systems and utilities as implemented at Master Disk Revision Level 21.0 (Rev. 21.0).

PRINTING HISTORY

CREDITS

HOW TO ORDER TECHNICAL DOCUMENTS

To order copies of documents, or to obtain a catalog and price list:

United States Customers

Call Prime Telemarketing,
toll free, at 1-800-343-2533,
Monday through Friday,
8:30 a.m. to 5:00 p.m. (EST).

International

Contact your local Prime
subsidiary or distributor.

CUSTOMER SUPPORT

Prime provides the following toll-free numbers for customers in the United States needing service:

1-800-322-2838 (within Massachusetts)
1-800-541-8888 (within Alaska)
1-800-651-1313 (within Hawaii)
1-800-343-2320 (within other states)

For other locations, contact your Prime representative.

SURVEYS AND CORRESPONDENCE

Please comment on this manual using the Reader Response Form provided in the back of this book. Address any additional comments on this or other Prime documents to:

Technical Publications Department
Prime Computer, Inc.
500 Old Connecticut Path
Framingham, MA   01701

This warning applies to the 9950, the 2250, and to all 50 Series processors manufactured after October 1, 1983

WARNING

This equipment generates and uses radio frequency energy and if not installed and used properly, i.e., in strict accordance with the instructions manual, may cause harmful interference to radio communications. It has been tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment.

Operation of this equipment in a residential area is likely to cause interference, in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

If there are any questions, please contact your Prime Field Office.

This warning applies to all other processors described in this book.

WARNING

This equipment generates and uses radio frequency energy and if not installed and used properly, i.e., in strict accordance with the instructions manual, may cause harmful interference to radio communications. As temporarily permitted by regulation, it has not been tested for compliance with the limits for Class A computing devices persuant to Subpart J of Part 15 of FCC rules, which are designed to provide reasonable protection against such interference.

Operation of this equipment in a residential area is likely to cause interference in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

If there are any questions, please contact your Prime Field Office.

CONTENTS

APPENDICES

ABOUT THIS BOOK

The System Administrator's Guide, Volume II: Communication Lines and Controllers is the second volume of the *System Administrator's Guide* set. This book provides definitions, instructions, and examples of the directives and commands necessary to configure communication lines. If you have administrative responsibility for a Prime system, this book is intended for you. Other administrative functions are described in the remaining books in the set:

- *System Administrator's Guide, Volume I: System Configuration* (DOC10131-1LA) discusses planning your configuration, installation issues, and the responsibilities of the System Administrator. It contains a master dictionary of all configuration directives, an overview of all the PRIMOS® directories and files, and a template for the PRIMOS.COMI file.

- *System Administrator's Guide, Volume III: System Access and Security* (DOC10133-1LA) documents all the security features available on the operating system, including ACLs, EDIT_PROFILE, and the Security Audit facility. It also describes environmental factors and orderly procedures necessary to maintain the security of terminals, peripherals, and storage media.

You are expected to have some familiarity with Prime systems before reading the volumes of the System Administrator's Guide. If you are not familiar with the PRIMOS operating system, read the *Prime User's Guide* (DOC4130-4LA) and its Rev. 20.0 update package (UPD4130-41A). These documents explain Prime's file management system and describe essential commands and utilities.

Contents

This book contains the following chapters and appendices:

Chapter 1, Communication Controllers: Describes the relationship between the backplane and communication controllers and discusses the electrical interfaces supported by Prime. This chapter identifies the AMLC, ICS, LHC, and LTS controllers, lists each controller's interface standards, and describes the features supported by each.

Chapter 2, Operating Intelligent Controllers: Introduces the COMM_CONTROLLER command that enables you to downline load protocols and runfiles to ICS, LHC, or LTS controllers interactively and at cold start.

Chapter 3, Useful Configuration Directives: Provides a reference table of all the configuration directives currently supported and a dictionary of all the directives mentioned in this book.

Chapter 4, Configuring Asychronous Lines: Provides a detailed discussion of both interactive and privileged use of the SET_ASYNC command and the CONVERT_AMLC_COMMANDS utility. It briefly discusses asynchronous communication, outlines the procedure for configuring asynchronous lines, and explains how to determine line numbers and enable Auto Speed Detect.

Chapter 5, Assignable Asynchronous Lines: Explains how to use the ASSIGN ASYNC and UNASSIGN ASYNC commands to assign and release local assignable lines. The second part explains how to use the NTS_ASSOCIATE, NTS_UNASSOCIATE, and NTS_LIST_ASSOCIATE commands to assign, display, and release NTS assignable lines. A brief discussion of serial communication and a list of Prime products that require assignable lines are included.

Chapter 6, Allocating I/O Buffers: Describes asynchronous character processing in PRIMOS, I/O interrupt rates, I/O buffers, default buffer assignments, and methods of calculating buffer capacity.

Appendix A, Obsolete and Rarely Used Commands: Provides a complete discussion of the AMLC command, ASSIGN AMLC, and UNASSIGN AMLC; discusses constructing octal bitstrings and offers a new set of fill-in-the-blank templates for the config word and lword arguments.

Appendix B, Determining Physical Line Numbers: Describes the procedure for tracing communications lines back to the controller. A new table shows how AMLC controllers map physical lines to logical line numbers and user number.

Related Documentation

Other helpful Prime documention includes the following:

- *ICS User's Guide* (DOC10094-1LA) and its update package for Rev. 21.0 (UPD10094-11A) provide detailed information on the Prime Model 2 (ICS2) and Model 3 (ICS3) Intelligent Communications Subsystems.

- *Operator's System Overview* (DOC9298-2LA) introduces the series of operator's guides and describes computer-room operation of Prime systems.

- *Operator's Guide to System Monitoring* (DOC9299-3LA) describes how to monitor system activity and respond to system and user messages.

- *Operator's Guide to File System Maintenance* (DOC9300-3LA) describes the PRIMOS file system and explains how to format partitions with MAKE, run the disk maintenance program FIX_DISK, determine physical device numbers, and interpret disk error messages.

- *Operator's Guide to System Backups* (DOC9301-1LA) and its update packages for Rev. 20.0 (UPD9301-11A and UPD9301-12A) and Rev. 20.2 (UPD9301-13A) describe how to save information on disk or tape, and how to restore that information later.

- *Operator's Guide to the Batch Subsystem* (DOC9302-3LA) describes how to use the Batch Subsystem.

- *Operator's Guide to System Commands* (DOC9304-3LA) details the commands used by system operators.

- *Operator's Guide to the Spooler Subsystem* (DOC9303-2LA describes how to set up, monitor, and control the Spooler subsystem.

- *PRIMOS Commands Reference Guide* (DOC3108-6LA) references all user commands in detail.

- *Site Preparation Guide* (DOC5029-3LA) provides information for preparing and maintaining a system site.

Network Documentation Available at Rev. 21.0

- *PRIMENET Planning and Configuration Guide* (DOC7532-3LA)

- *Programmer's Guide to Prime Networks* (DOC10113-1LA)

- *Operator's Guide to Prime Networks* (DOC10114-1LA)

- *User's Guide to Prime Network Services* (DOC10115-1LA)

- *NTS Planning and Configuration Guide* (DOC10159-1LA)

- *NTS User's Guide* (DOC10117-1LA)

- *WSI300 User's Guide* (DOC10155-1LA)

PRIME DOCUMENTATION CONVENTIONS

The following conventions are used in command formats, statement formats, and in examples throughout this document. Examples illustrate the uses of these commands and statements in typical applications.

Convention	*Explanation*	*Example*
UPPERCASE	In command formats, words in uppercase indicate the names of commands, options, statements, and keywords. Enter them in either uppercase or lowercase.	SLIST
lowercase	In command formats, words in lowercase indicate variables for which you must substitute a suitable value.	LOGIN user-id
Abbreviations in format statements	If an uppercase word in a command format has an abbreviation, either the abbreviation is underscored or the name and abbreviation are placed within braces.	LOGOUT $\left\{ \begin{array}{l} \text{SET\_QUOTA} \\ \text{SQ} \end{array} \right\}$
Brackets [ ]	Brackets enclose a list of one or more optional items. Choose none, one, or more of these items.	LD $\left[\begin{array}{l} \text{-BRIEF} \\ \text{-SIZE} \end{array} \right]$
Braces { }	Braces enclose a list of items. Choose one and only one of these items.	CLOSE $\left\{ \begin{array}{l} \text{filename} \\ \text{ALL} \end{array} \right\}$
Hyphen -	Wherever a hyphen appears as the first character of an option, it is a required part of that option.	SPOOL -LIST
Underscore in examples	In examples, user input is underscored but system prompts and output are not.	OK, RESUME MY_PROG This is the output of MY_PROG.CPL
Subscripts	A subscript after a number indicates that the number is not in base ten. For example, the subscript 8 is used for octal numbers.	200_8
Angle brackets in messages < >	In messages, text enclosed within angle brackets indicates a variable for which the program substitutes the appropriate value.	Disk <diskname>

New Features At Rev. 21.0

Rev. 21.0 introduces several new commands, communication protocols, and configuration directives, as well as enhancements to existing commands, as follows:

- Prime ECS protocols TTY8 and TTY8HS

- The ASSIGN ASYNC command

- The COMM_CONTROLLER command

- The NTS_ASSOCIATE command

- The NTS_LIST_ASSOCIATE command

- The NTS_UNASSOCIATE command

- The UNASSIGN ASYNC command

- The COMDVM directive

- The LHC directive

- The MIRROR directive

- The NTSABF directive

- The NTSASL directive

- The NTSBUF directive

- The NTSUSR directive

- The PAGING directive

- The PAGINM directive

- SET_ASYNC command enhancements:

      User SET_ASYNC
      Prime ECS support
      NTS support
      System defaults
      Cold start defaults
      Display line characteristics
      Per-line dislog
      Auto Speed Detect

COMMUNICATION CONTROLLERS

Although users cannot communicate directly with the CPU, high speed communication controllers create that illusion. Communication controllers are printed circuit boards, some with resident microprocessors, that are located in the backplane. They pass data, control signals, and address information between the CPU and user terminals, printers, and other peripheral devices.

The first part of this chapter describes Prime communication controllers in general. It discusses the backplane, electrical interface standards, configuration limits, and the STAT COMM command.

The second part of this chapter describes Prime communication controllers in detail. It discusses device address assignments, asynchronous line support, and specific controller characteristics.

The Backplane

The term **backplane** describes the chassis in the Prime computer system that holds the various printed circuit boards that make up a Prime computer.

The backplane connects all the boards in a system, acts as a source of DC power, and provides a common interface between the CPU, memory, and the communication controllers. I/O busses built into the backplane provide separate interfaces for data, addresses, and control signals.

The standard backplane has sufficient slots for installing the CPU, memory, communication controllers, standard peripheral subsystems, and power supplies. Figure 1-1 is a diagram of the standand Prime backplane. It shows the common location of these printed circuit boards.

The total number of boards plugged into the backplane is limited only by the number of slots and the amount of amperage available. The exact choice depends on your system's specific requirements.
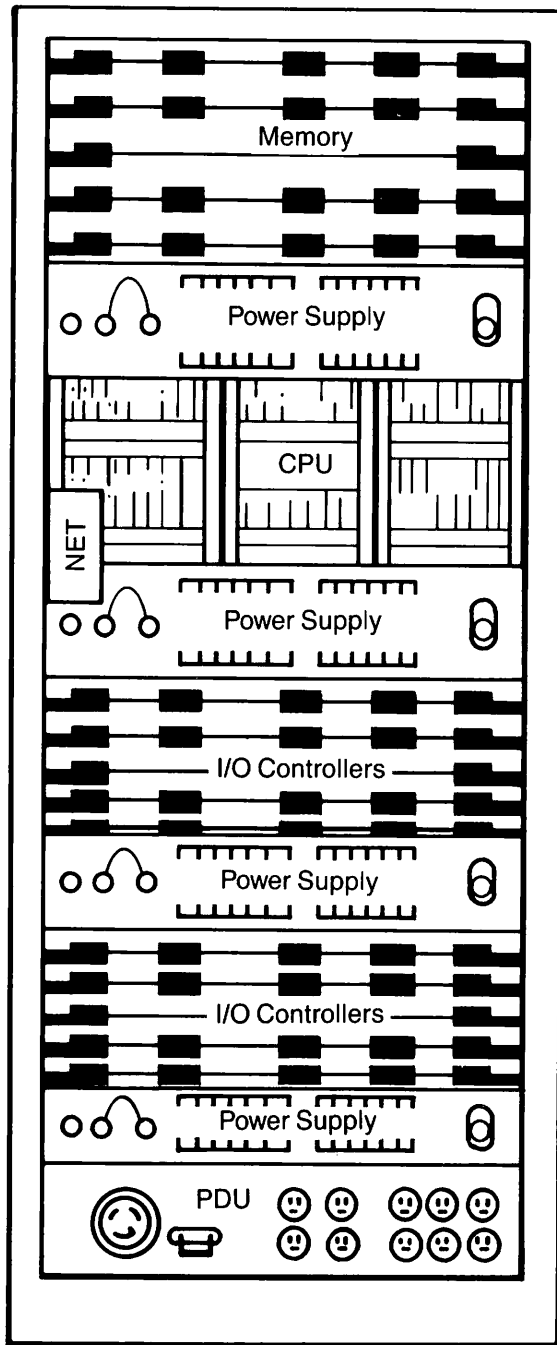
FIGURE 1-1. The Standard Prime Backplane

Communication controllers are also known as I/O controllers. Figure 1-1 shows the CPU cabinet from the back. The diagram is intended to show the relative position rather than the exact location of the printed circuit boards.

Memory boards are commonly installed in the top of the backplane supported by their own power supply. The CPU boards usually are installed below the memory array and also have their own power supply. Depending on your installation and system configuration limits, you can have one or two groups of communication controllers. If the total amperage requirements for all the communication controllers exceeds the capacity of one power supply, you can install another. The Power Distribution Unit is usually located at the base of the CPU cabinet.

There are several types of printed circuit boards in the backplane. A typical system can include a selection of the following boards:

- Asynchronous MultiLine Controllers (AMLC)
- Burst Mode Disk Controllers (BMSMC)
- Burst Mode Tape Controllers (BMTC)
- DC Power supplies
- Floppy Disk Controller
- High Speed Synchronous MultiLine Controllers (HSSMLC)
- Intelligent Communication Subsystem Controllers (ICS)
- Interprocess Communication Controller (IPC)
- LAN Host Controllers (LHC)
- Magnetic Tape Controllers (MTC)
- Memory Array (MA) Boards
- Multiple Data Link Controllers (MDLC)
- Power Distribution Unit (PDU)
- PRIMENET™ Node Controller (PNC)
- SMD, CMD, and FMD Disk Controllers
- Synchronous MultiLine Controllers (SMLC)
- System Option Controllers (SOC)
- The Cental Processing Unit (CPU)
- Unit Record Controllers (URC)
- Virtual Control Pannel (VCP)

Electrical Interface Standards

An interface is a standard method of interpreting the electrical signals emitted from the pins in the connectors of a data communication line. When you connect terminals, modems, or any other electronic devices to a host, an interface ensures that

● Voltage and signal levels are compatible.

● Interface connectors plug together with identical pin wiring and corresponding pin connections.

● Control information supplied by one device is understood by the other device.

Prime communications controllers support three communications interface standards:

● 20 milliamp current loop

● IEEE 802.3

● EIA RS-232-C (also called CCITT V.24 in Europe)

20 Milliamp Current Loop

The 20 milliamp current loop is a communications interface that supports devices that operate at relatively slow baud rates. This interface is available only on AMLC controllers. PT25™, PT45™, and PT65™ terminals can use either 20 milliamp current loop or RS-232-C/CCITT V.24 lines.

IEEE 802.3

The IEEE 802.3 defines the digital interface for asynchronous local area networks. This interface is available only on Prime LAN Host Controllers and LAN Terminal Servers running Network Terminal Service (NTS) software.

EIA RS-232-C/CCITT V.24

EIA RS-232-C/CCITT V.24 is an industrywide communications standard that describes a communications interface. It is commonly used for modems, terminals and printers, including all Prime products that use asynchronous lines. RS-232-C is the standard adopted by manufacturers in the U.S. and Canada; CCITT V.24 is the equivalent European standard.

RS-232-C defines the signals assigned to particular pin numbers in a 25-pin connector. Although the international standard uses all 25 pins, not all devices use the same pin connections, timing, control characters, or programming procedures.

Prime synchronous protocols support the full 25-pin data set support shown in Table 1-1. Prime asynchronous protocols support a subset. For example, transmit and receive timing for dialup lines is controlled by the modem's internal clock rather than by pins 15 and 17.

For device-specific information on other vendor's hardware, refer to the manufacturer's installation guide.

TABLE 1-1. EIA RS-232-C/CCITT V.24 Electrical Interface

Pin	Signal Definition
1	Protective ground
2	Transmit data
3	Receive data
4	Request to send
5	Clear to send
6	Data set ready
7	Signal return (ground)
8	Data carrier detect
9	DC+ test voltage
10	DC- test voltage
11	Unassigned
12	Second data carrier detect
13	Second clear to send
14	Secondary transmit data
15	Transmit clock
16	Secondary receive data
17	Receive clock
18	Unassigned
19	Secondary request to send
20	Data terminal ready
21	Signal quality detector
22	Ring indicator
23	Data signal rate selector
24	DTE transmit signal
25	Unassigned

System Configuration Limits

The maximum number of controller boards that can be installed depends on the type of system that you have. This section outlines each system's configuration limits. For more information, see your CPU handbook.

The 2250 Backplane

The 2250™ system has a 10-slot backplane. The 2-board CPU requires 3 slots because the center slot must remain empty for ventilation. The 1MB memory array uses 2 slots. The combined disk and tape controller uses 1 slot. A 1-slot streamer tape controller is also available. 2250 systems are equipped with an ICS1 controller that occupies 1 slot. The ICS1 is required on a 2250 because it contains the hardware logic for the I/O bus. If additional asynchronous lines are desired, you can add AMLC or ICS controllers.

The 2350 Backplane

The 2350™ system has a 9-slot backplane. The 2-board CPU requires 3 slots because the center slot must remain empty for ventilation. The 2350 contains a 1-slot diagnostic processor. The 2MB or 4MB memory array uses 1 slot. The combined disk and tape controller uses 1 slot. 2350 systems are equipped with either an ICS1 or AMLC controller that occupies 1 slot.

The 2450 Backplane

The 2450™ system has a 9-slot backplane. The 2-board CPU requires 3 slots because the center slot must remain empty for ventilation. The 2450 contains a 1-slot diagnostic processor. The 2MB or 4MB memory array uses 1 slot. The combined disk and tape controller uses 1 slot. 2450 systems are equipped with an AMLC controller that occupies 1 slot.

The 2750 Backplane

The 2750™ system has a 16-slot backplane. The system is available with several different memory, disk, and tape configurations and supports a 2-board CPU. There are up to 7 slots available for controllers.

The 6350 Backplane

The 6350™ system has a twin-bay cabinet. The 6350 CPU and memory use their own backplane and power supplies and do not affect the I/O configuration of the system. The 6350 supports up to 13 controllers.

The 6550 Backplane

The 6650™ system has a twin-bay cabinet. The 6650 CPU and memory use their own backplane and power supplies and do not affect the I/O configuration of the system. The 6650 supports up to 22 controllers.

The 9755 Backplane

The 9755™ system has a twin-bay cabinet and comes equipped with a CPU, diagnostic processor, one or two 8MB memory boards, one disk subsystem, and power supplies. The system is available with several different packages. Up to 10 slots are available for communication controllers.

The 9955II Backplane

The 9955II™ system has a twin-bay cabinet and comes equipped with a CPU, diagnostic processor, one or two 8MB memory boards, one disk subsystem, and power supplies. The system is available with several different packages. Up to 14 slots are available for controllers.

Communication Controller Configuration Limits

Communication controllers for the 50 Series™ systems are compatible. You can install any combination of communication controllers up to the maximum number allowed provided you have available slots in the backplane, sufficient amperage, and enough bulkhead connector space.

Synchronous Communication Support

Prime ICS, MDLC, SMLC, and HSSMLC controllers provide synchronous communication support. Synchronous data transmission transfers data in blocks, which are framed by start and end-of-text characters. Systems that transfer a large volume of data at high speeds typically use synchronous data transmission. You can have a maximum of eight synchronous lines on a system regardless of which controllers you use.

Asynchronous Communication Support

Prime AMLC, ICS, and LHC controllers provide asynchronous communication support. Asynchronous data transmission transfers characters individually. Each character is framed with start and stop bits. Systems that support a large number of interactive terminal users typically use asynchronous data transmission.

Table 1-2 shows the maximum number of communication controllers supported at PRIMOS Rev. 21.0 and their line loads.

TABLE 1-2. PRIMOS Rev. 21.0 Communication Controller Configuration Limits

Controller Load	Line Load and Restrictions
8 AMLC	Each controller supports a maximum of 16 asynchronous lines.
4 ICS	Any combination of ICS1, ICS2, and ICS3 controllers.
	ICS1 supports a maximum of 8 asynchronous and 1 full-duplex synchronous line (optional).
	ICS2 supports a 16-LAC card cage containing any combination of 4-line asynchronous LACs and as many as four 2-line synchronous LACs, provided that the maximum line load is not exceeded.
	ICS3 supports an 8-LAC or 16-LAC card cage containing any combination of 4-line asynchronous LACs and as many as four 2-line synchronous LACs, provided that the maximum line load is not exceeded.

The maximum line load for ICS2 and ICS3 is

64	ASYNC
8	FDX SYNC SNA
8	FDX SYNC RJE
3	HDX SYNC X.25
3	FDX SYNC X.25

Controller Load	Line Load and Restrictions
6 LHC	Each controller supports a maximum of 128 asynchronous NTS lines, or 32 asynchronous lines when PRIMENET is also running on the same LHC. The maximum configuration is 4 LHCs for NTS and 2 LHCs for PRIMENET.
2 MDLC	Each controller supports 2 or 4 synchronous lines for FDX, HDX PRIMENET, or RJE with on-board firmware support for as many as two protocols.
1 PNC	Each controller supports local ring network with PRIMENET software.
2 HSSMLC	Each controller supports 2 or 4 synchronous lines with on-board firmware support for one protocol.
2 URC	Each controller supports a maximum of 2 line printers and a card reader.

Monitoring Your Controllers

The STATUS COMM command provides an overview of your communication lines and controllers. Use the STATUS COMM command to display the line count, device name, and octal device address for all the controllers connected to your system. You can issue the STAT COMM command from the supervisor terminal or any user terminal. If any of these devices are identified by model (LHC), type (AMLC), downline load file number (ICS), a PROM set ID number (MDLC), or firmware option number (SMLC), this information is displayed. Otherwise, the *type* field is blank.

Each controller's total line count is calculated separately for synchronous and asynchronous lines. Line counts for inoperable lines are calculated and displayed in the same manner. An entry in the *bad-lines* field indicates that a line is inoperable. A line is listed as inoperable when the software configuration, specifically the ICS CARDS directive, does not agree with the actual configuration or when the hardware is broken. This is a recoverable problem; other lines attached to the same LAC continue to operate.

The following example of the STATUS COMM command shows eight communication controllers. One is an ICS2 controller, using downline load file ICS2_01.DL, at device address 10_8. It currently supports two synchronous lines and 60 asynchronous lines, one of which is inoperable.

```
OK, STAT COMM
```

Controller	Type	Device Address	Total-Lines Async	Total-Lines Sync	Bad-Lines Async	Bad-Lines Sync
ICS2	F-01	10	60	2	1	0
LHC	300	11	No Information		No Information	
ICS1		37	8	1	0	0
MDLC	5646	50	0	4	No Information	
AMLC	DMQ	52	16	0	No Information	
AMLC	DMQ	53	16	0	No Information	
AMLC	DMQ	54	16	0	No Information	
LHC	300	56	No Information		No Information	

The following example shows 4 asynchronous controllers that support 64 communication lines. This system has no synchronous controllers.

```
OK, STAT COMM
```

Controller	Type	Device Address	Total-Lines Async	Total-Lines Sync	Bad-Lines Async	Bad-Lines Sync
AMLC	DMQ	35	16	0	No Information	
AMLC	DMQ	52	16	0	No Information	
AMLC	DMQ	53	16	0	No Information	
AMLC	DMQ	54	16	0	No Information	

Device Addresses

Device addresses are two-digit octal numbers that identify every controller. The CPU uses device addresses to direct instructions and data to the correct device. For certain asynchronous controllers, device addresses determine the sequence of line number allocation.

Device addresses are not determined by physical location in the backplane; each controller has a unique device address assigned by the manufacturer. Most device addresses are hardwired on the board in the factory, but some device addreses are set with dip switches when the controller is installed.

In systems with identical model controllers, the device addresses must be different so that the CPU can recognize and communicate with the correct controller. Table 1-3 lists the valid address assignments for communication controllers.

Note

PRIMOS performs a complete software check at cold start. If you supply an incorrect device address for the ICS CARDS, LHC, or SYNC configuration directives or for the COMM__CONTROLLER command, an error message is displayed at the supervisor terminal.

TABLE 1-3. Valid Device Addresses Assignments

Address	*Communication Controller*
00_8	Unused
01_8	Paper tape reader
02_8	Paper tape punch
03_8	First Unit Record Controller (URC) (line printer, card reader, card punch)
04_8	Supervisor terminal
05_8	Second Unit Record Controller (URC) (line printer, card reader, card punch)
06_8	InterProcess Communication board (IPC)
07_8	PRIMENET Node Controller (PNC)
10_8	First Intelligent Communication Subsystem (ICS1, ICS2, ICS3) LAN Host Controller (LHC)
11_8	Second Intelligent Communication Subsystem (ICS1, ICS2, ICS3) LAN Host Controller (LHC)

TABLE 1-3. Valid Device Addresses Assignments (continued)

Address	Communication Controller
12_8	Floppy Disk Controller
13_8	Second Magnetic Tape Controller (MTC)
14_8	First Magnetic Tape Controller (MTC)
15_8	Fifth Asynchronous MultiLine Controller (AMLC) Intelligent Communication Subsystem 1 (ICS1) LAN Host Controller (LHC)
16_8	Sixth Asynchronous MultiLine Controller (AMLC) Intelligent Communication Subsystem 1 (ICS1) LAN Host Controller (LHC)
17_8	Seventh Asynchronous MultiLine Controller (AMLC) Intelligent Communication Subsystem 1 (ICS1) LAN Host Controller (LHC)
20_8	Virtual Control Panel (VCP) Real Time Clock (RTC) System Option Controller (SOC)
21_8	Disk Option B'
22_8	Fourth disk controller
23_8	Eighth disk controller
24_8	First disk controller Writable control storage
25_8	Third disk controller Disk Option B
26_8	Second disk controller
27_8	Sixth disk controller
30_8	First buffered parallel I/O channel
31_8	Second buffered parallel I/O channel
32_8	Eighth Asynchronous MultiLine Controller (AMLC) Intelligent Communication Subsystem 1 (ICS1) LAN Host Controller (LHC)

TABLE 1-3. Valid Device Addresses Assignments (continued)

Address	Communication Controller
33_8	First 3008-3009 printer plotter
34_8	Second 3008-3009 printer plotter
35_8	Fourth Asynchronous MultiLine Controller (AMLC) Intelligent Communication Subsystem 1 (ICS1) LAN Host Controller (LHC)
36_8	Third Intelligent Communication Subsystem (ICS1, ICS2, ICS3) LAN Host Controller (LHC)
37_8	Fourth Intelligent Communication Subsystem (ICS1, ICS2, ICS3) LAN Host Controller (LHC)
40_8	PRIMAD Analog Input System (AIS)
41_8	First Digital Input System (DIS)
42_8	Second Digital Input System (DIS)
43_8	First Digital Output System (DOS)
44_8	Second Digital Output System (DOS)
45_8	Fifth disk controller Digital to Analog Output System
46_8	Computer products interface
47_8	Seventh disk controller PRIMENET Node Controller (PNC)
50_8	First Multiple Data Link Controller (MDLC) High Speed Synchronous Multiline Controller (HSSMLC) LAN Host Controller (LHC)
51_8	Second Multiple Data Link Controller (MDLC) High Speed Synchronous Multiline Controller (HSSMLC) LAN Host Controller (LHC)
52_8	Third Asynchronous MultiLine Controller (AMLC) Intelligent Communication Subsystem 1 (ICS1) LAN Host Controller (LHC)

TABLE 1-3. Valid Device Addresses Assignments (continued)

Address	Communication Controller
53_8	Second Asynchronous MultiLine Controller (AMLC) Intelligent Communication Subsystem 1 (ICS1) LAN Host Controller (LHC)
54_8	First Asynchronous MultiLine Controller (AMLC) Intelligent Communication Subsystem 1 (ICS1) LAN Host Controller (LHC)
55_8	Multiple Autocall system
56_8	Synchronous Multiline Controller (SMLC) LAN Host Controller (LHC)
57_8	Unused
60_8	General purpose interface board
61_8	General purpose interface board
62_8	General purpose interface board
63_8	General purpose interface board
64_8	General purpose interface board
65_8	General purpose interface board
66_8	General purpose interface board
67_8	General purpose interface board
70_8	Reserved for specials
71_8	Reserved for specials
72_8	Reserved for specials
73_8	Reserved for specials
74_8	Reserved for specials
75_8	Reserved
76_8	Reserved
77_8	I/O bus tester

Asynchronous Line Support

The maximum number of asynchronous lines that can exist on a system is currently 256. One of these lines is reserved for the supervisor terminal. You can distribute the remaining 255 lines between LHC, AMLC, and ICS controllers.

An LHC controller can support as many as 128 asynchronous NTS connections at one time. If PRIMENET software is also running on the LHC the maximum number of asynchronous connections is 32.

Current models of AMLC controllers support a maximum of 16 lines each. You can install a maximum of 8 AMLC controllers in larger systems for a total of 128 lines.

Like AMLC controllers, ICS1 controllers are located in the backplane. Each ICS1 controller supports 1 synchronous (optional) and a maximum of 8 asynchronous lines.

ICS2 and ICS3 controllers have two basic parts: the controller board, which is located in the backplane; and an external card cage, which is located in a peripheral cabinet.

Card cages come in 8-slot models for 2350s and 2450s, and 16-slot models for all other systems. Each slot can hold one 4-line asynchronous Line Adapter Card (LAC) or one 2-line synchronous LAC to add more lines as your need for lines increases. Thus, an 8-slot cage can support a maximum of 32 asynchronous lines, and a 16-slot cage can support a maximum of 64 asynchronous lines. You can connect any combination of 8-slot and 16-slot card cages to your system provided that the total number of lines connected to your system does not exceed 256.

LACs are usually distributed among card cages. You do not have to load them into adjacent slots. An empty slot does not affect operation. ICS controllers recognize the presence of a LAC and allocate logical line numbers based on the order in which the cards appear. Logical line numbers are allocated first to lines attached to AMLC controllers, then to lines attached to ICS controllers based on the controller's address.

Line Number Allocation

Line numbers for NTS users are dynamically allocated when the connection is made and released when the user disconnects. NTS line numbers start at 1024. Line number allocation for AMLC and ICS controllers, however, is fixed. Line numbers are allocated first to lines attached to AMLC controllers, then to lines attached to ICS controllers based on the controller's address.

Table 1-4 shows the line number assignment to AMLC controllers. This assignment depends on the hardware and can not be changed. If an AMLC device address is skipped, the line numbers for that controller are still reserved.

TABLE 1-4.   Line Number Assignment for AMLC Controllers

Controller Number	Device Address	Line Numbers
First	54_8	0 to 15
Second	53_8	16 to 31
Third	52_8	32 to 47
Fourth	35_8	48 to 63
Fifth	32_8	64 to 79
Sixth	17_8	80 to 95
Seventh	16_8	96 to 113
Eighth	15_8	112 to 127

Lines attached to ICS controllers are assigned line numbers based on device address in the following order: 10_8, 11_8, 36_8, and 37_8.

ICS2 and ICS3 controllers are given a device address of 10_8 or 11_8 by the manufacturer. ICS1 controllers are usually assigned a device address of 36_8 or 37_8. However, if you have only ICS1 controllers on your system, or a mixture of ICS1 and AMLC controllers, you can use any valid asynchronous device address (10_8, 11_8, 15_8, 16_8, 17_8, 32_8, 35_8, 36_8, 37_8, 52_8, 53_8, and 54_8).

The following example shows three fully-loaded controllers. Blocks of line numbers have been allocated to each controller.

Controller Type	Device Address	Line Count	Line Numbers
AMLC	54_8	16 Lines	0 to 15
ICS3	10_8	32 Lines	16 to 47
ICS1	36_8	8 Lines	48 to 55

The first line number on each ICS controller is always a multiple of eight. After all the lines connected to an ICS controller receive line numbers, the line numbers skip to the next multiple of eight for the next ICS controller. This can result in line numbers being allocated for lines that are not currently in use.

In the example below, gaps occur from line numbers 44 to 47 and from 108 to 111. The block of line numbers from 0 to 15 is reserved for an AMLC controller at device address 54_8.

Controller Type	Device Address	Line Count	Line Numbers
AMLC	53_8	16 Lines	16 to 31
ICS3	10_8	12 Lines	32 to 43
ICS2	11_8	60 Lines	48 to 107
ICS1	36_8	8 Lines	112 to 119

AMLC Controller Characteristics

An Asynchronous MultiLine Controller (AMLC) is a hardwired, single-board communication controller. It responds to hardware defined instructions in the CPU called programmed I/O instructions, commonly known as PIO. An AMLC can provide CCITT V.24 RS-232-C or 20 milliamp current loop support for terminals, peripherals, and data sets. Figure 1-2 shows an AMLC controller board with FCC compliant cables.

AMLC controllers use Universal Asynchronous Receiver Transmitters (UARTs), one pair per line, to convert serial characters from devices to parallel characters for the CPU. UARTs append stop and start bits to each character before passing it to the Direct Memory Channel (DMC) tumble table, one per board. A special process known as the AMLDIM services the DMC tumble tables.

All AMLC boards are shipped from the factory with a device address of 54_8. If you have more than one AMLC board in your system, you must rewire the device address when the board is installed. Valid AMLC device addresses are 54_8, 53_8, 52_8, 35_8, 15_8, 16_8, 17_8, and 32_8.

An AMLC controller can support eight line speeds as follows:

- Four fixed speeds (110, 134.5, 300, and 1200 baud)

- One programmable speed set with the AMLCLK configuration directive (default 9600 baud)

- Three user-selectable speeds hardwired with on-board jumper cables (preset at the factory to 75, 150, and 1800 baud)

Note

You can change the user-selectable speeds when the controller is installed. Ask your Customer Support Center to rewire the board and change your ASYNC JUMPER configuration directive to agree with the new speeds.

Current AMLC boards support 16 asynchronous communication lines. Older AMLC boards can support 8 or 16 lines depending on the specific model as shown in Table 1-5. The maximum number of AMLC controllers that can be installed in an FCC compliant cabinet is 4. Older model non-compliant cabinets can house 8. If additional local lines are desired, you can install an ICS controller.

Note

The baud rate of the last line on the last AMLC controller sets the AMLC I/O interrupt rate. For more information on interrupt rates and character processing, see Chapter 6, Allocating I/O Buffers.
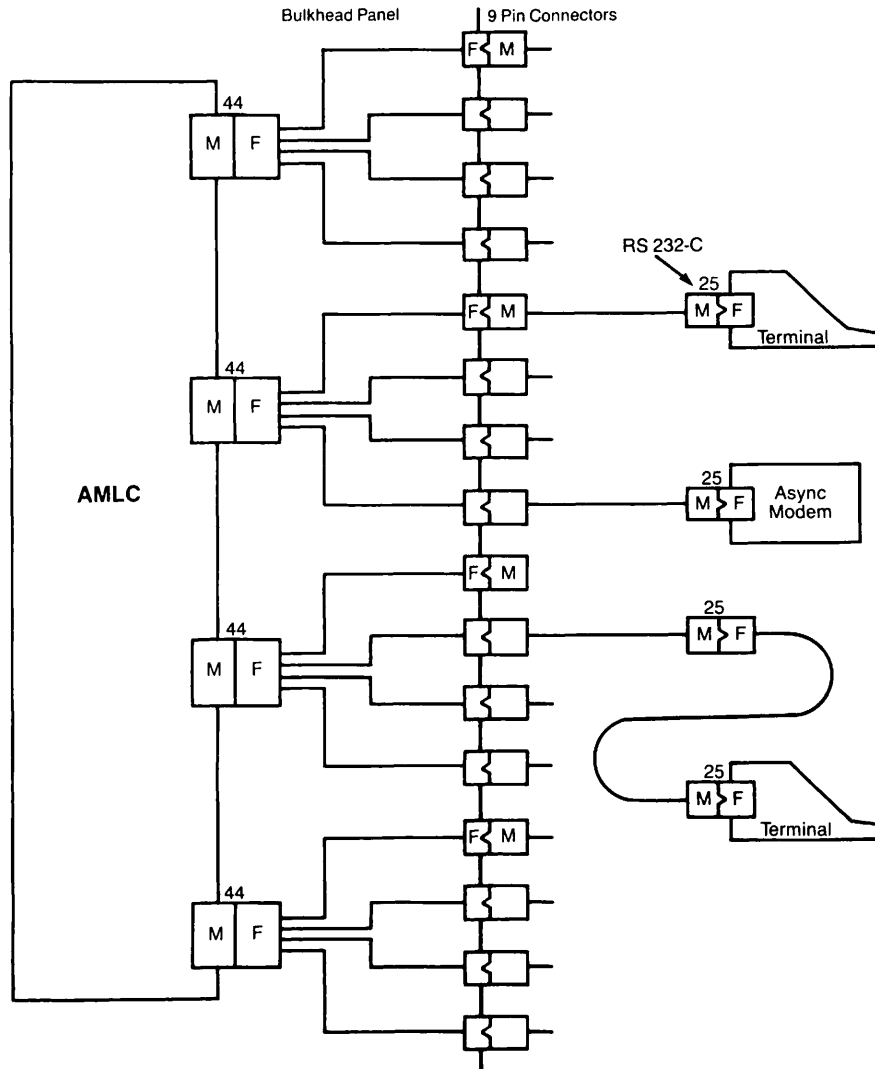
FIGURE 1-2. AMLC Board With FCC Compliant Cables

Table 1-5 summarizes the characteristics of AMLC controllers.

TABLE 1-5. Characteristics of AMLC Controllers

Controller	Lines	Comments
AMLC 5152	8 asynchronous	No longer marketed, but still supported by Prime
AMLC 5154	16 asynchronous	RS-232-C/CCITT V.24
AMLC 5174	16 asynchronous	20 milliamp current loop
AMLC 5174	16 asynchronous	Lines 0-7 20 milliamp Lines 8-15 RS-232-C/V.24

ICS Controller Characteristics

Intelligent Communication Subsystem Models ICS1, ICS2, and ICS3 are programmable communication controllers with on-board dual microprocessors. The microprocessors govern timing, initialization, configuration, and protocol processing. In this way, ICS controllers are able to reduce some of the overhead formerly generated by processes in the CPU that managed AMLC controllers. ICS controllers provide full CCITT V.24 RS-232-C support.

You can initialize, shutdown, and downline load runfiles to an ICS controller with the COMM_CONTROLLER command interactively or at cold start. For more information, see Chapter 2, Operating Intelligent Controllers.

ICS controllers use Direct Memory Queue (DMQ) input and output buffers to reduce the amount of CPU overhead involved in transmitting characters. The ICS INPQSZ configuration directive determines the capacity of the DMQ input buffers. A special process known as the ASYNDM services the DMQ buffers. ICS controllers have an internal timing mechanism that operates independently of individual lines or specific controllers. The ICS INTRPT configuration directive determines how frequently the ASYNDM interrupts the CPU. For more information on configuration directives, see Chapter 3, Useful Configuration Directives. For more information on character processing and interrupt rates, see Chapter 6, Allocating I/O Buffers.

Valid device addresses for ICS1 controllers are 10_8, 11_8, 36_8, and 37_8. If you have only AMLC and ICS1 controllers, however, you can use any available AMLC address for the ICS1 board.

Device addresses for ICS2 and ICS3 controllers are preset to 10_8 or 11_8. If you have more than two of either model, you can move the dip switches on the controller board when it is installed to 36_8 or 37_8.

The ICS1 is a single-board controller, which supports 8 asynchronous and 1 optional synchronous communication lines. ICS2 and ICS3 controllers consist of a controller board located in the backplane and an external line adapter card cage. Figure 1-3 shows an ICS1 with FCC compliant cables.

Card cages come in 8-LAC and 16-LAC versions. Each LAC can support either 2 synchronous lines or 4 asynchronous EIA RS-232-C communication lines. Figure 1-4 shows the cable connections for asynchronous ICS2 and ICS3 LACs. Each LAC contains a pair of UARTs for each line to perform serial data conversion and recognize packet framing. Dual baud rate generators residing on each LAC govern the transmit and recieve timers. For more information on ICS controllers, see the *ICS User's Guide.*
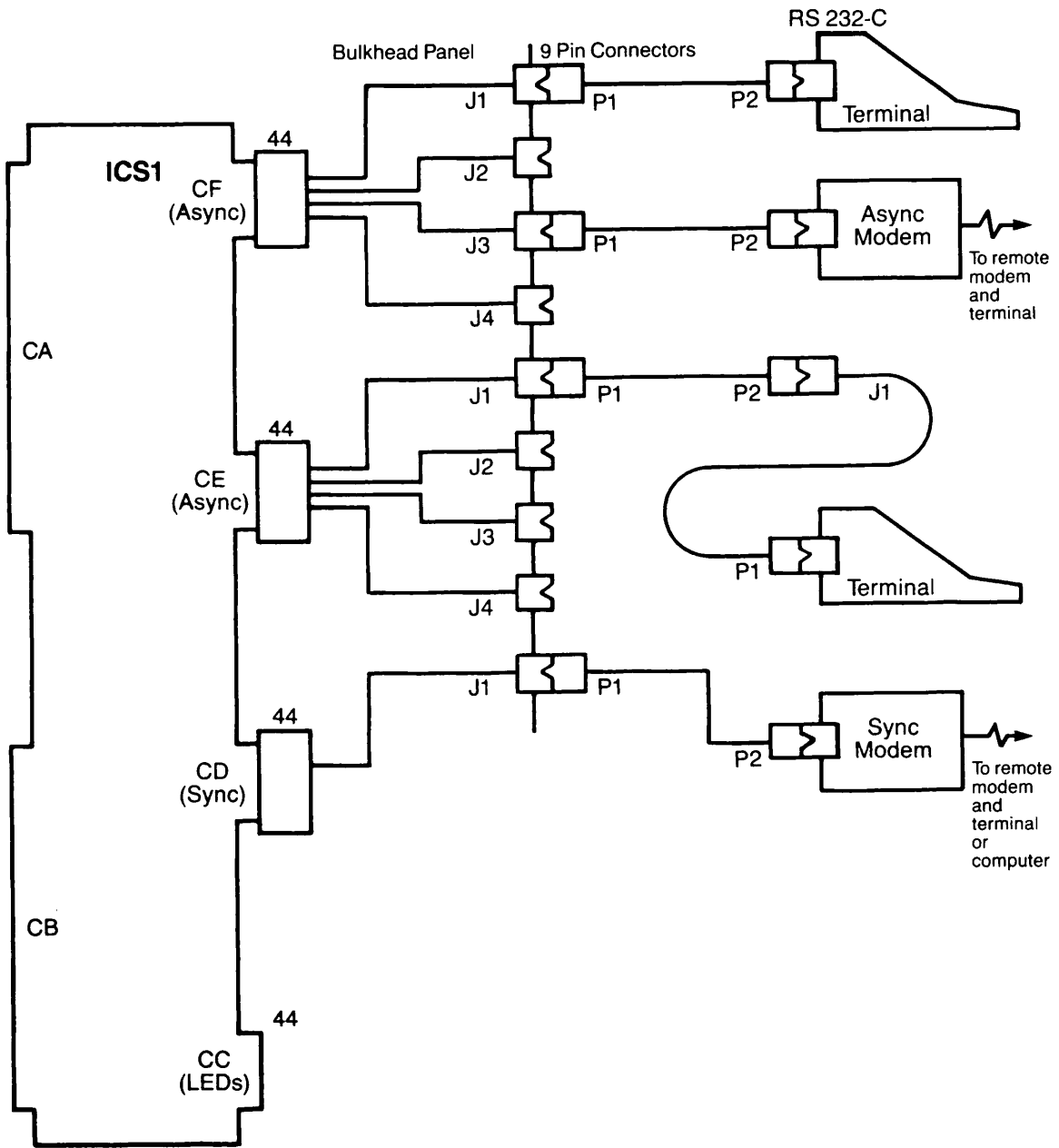
FIGURE 1-3. ICS1 Board With FCC Compliant Cables
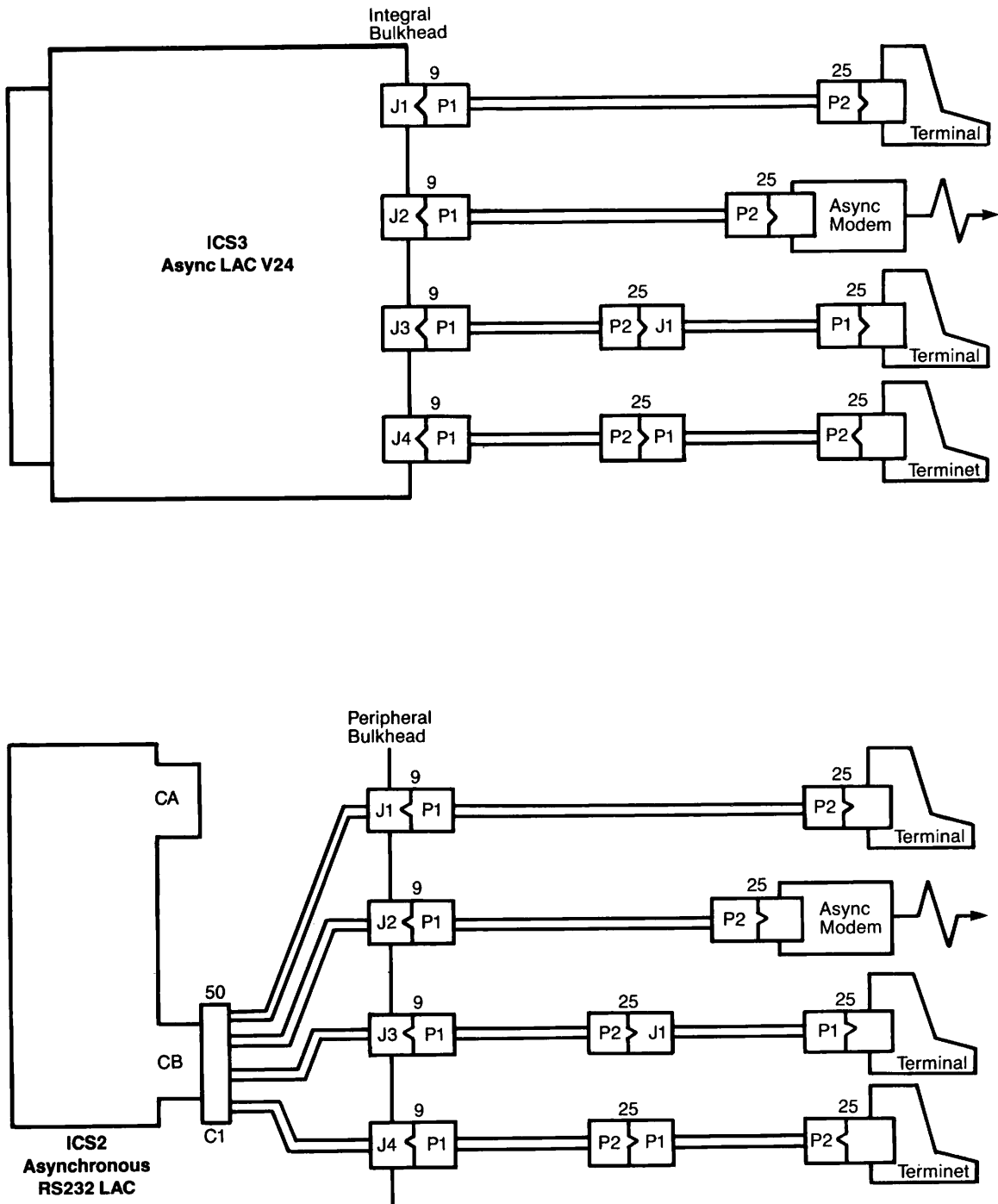
FIGURE 1-4.   Cable Connections for ICS2 and ICS3 Asynchronous LACs

Table 1-6 summarizes the characteristics of ICS controllers.

TABLE 1-6.   Characteristics of ICS Controllers

Controller	Maximum Lines	Comments
ICS 5181	8 asynchronous 1 synchronous (fixed)	RS-232-C/CCITT V.24 Synchronous line supports RJE (HASP/2780,3780) and X.25 FDX HDLC PRIMENET.
ICS 5242	64 asynchronous 8 synchronous	RS-232-C/CCITT V.24 Supports 16 slots for 2-line sync or 4-line async line adapter cards.
ICS2 5720	32 asynchronous 8 synchronous	RS-232-C/CCITT V.24 Provides async and sync SDLC SNA support V.35/DDS.
ICS3 5725	64 asynchronous 8 synchronous	RS-232-C/CCITT V.24 Basic model supports 16 slots for 2-line sync or 4-line async line adapter cards. Not for use in tower systems or 7564S cabinets.
ICS3 5730	32 asynchronous 8 synchronous	RS-232-C/CCITT V.24 Supports 8 slots for 2-line sync or 4-line async line adapter cards. For use in tower systems.
ICS3 5732	64 asynchronous 8 synchronous	RS-232-C/CCITT V.24 Supports 16 slots for 2-line sync or 4-line async line adapter cards in 6000 series systems only.
ICS3 5735	64 asynchronous 8 synchronous	RS-232-C/CCITT V.24 Supports 16 slots for 2-line sync or 4-line async line adapter cards. Contains an integral power supply.

Maintaining ICS Integrity

ICS card cages contain a maximum of 16 line adapter cards, each supporting four asynchronous or two synchronous lines. Use the optional ICS CARDS directive to have PRIMOS report, at cold start, any inconsistencies between the expected LAC configuration and the actual configuration. For more information on this directive, see Chapter 3, Useful Configuration Directives.

If, at any other time, you suspect that one or more LACs have failed,

1. Warm start the system.

2. If the warm start does not identify or resolve the problem, use the STATUS COMM command and record the results, making note of the number of lines attached to each ICS2 or ICS3 controller.

3. Cold start the system. If the ICS2 or ICS3 subsystem can identify a bad LAC, one of the following messages is displayed at the supervisor terminal:

   ```
   ICS device <dd>: async line <ee> (Jn) on line card in slot <y> is inoperable.

   ICS device <dd>: line card in slot <y> is inoperable.

   ICS device <dd>: line card in slot <y> is unrecognizable.
   ```

   dd identifies the device address, usually 10 or 11.

   ee identifies the physical line number.

   n identifies one of four jacks on the LAC.

   y identifies the affected slot on the controller.

4. If no message appears but the problem has not been resolved, use the STATUS COMM command again. Compare its output with that in step 2. If the output differs, a problem with at least one LAC is causing configuration discrepancies. Contact your Customer Support Center for help.

Changing Line Adapter Cards

If it becomes necessary to replace, remove, or add a LAC to the card cage, shut down PRIMOS and power off the system before doing so. After the changes are complete, cold start the system. Figure 1-5 shows an 8-slot ICS3 card cage loaded with one 2-line synchronous LAC and one 4-line asynchronous LAC.
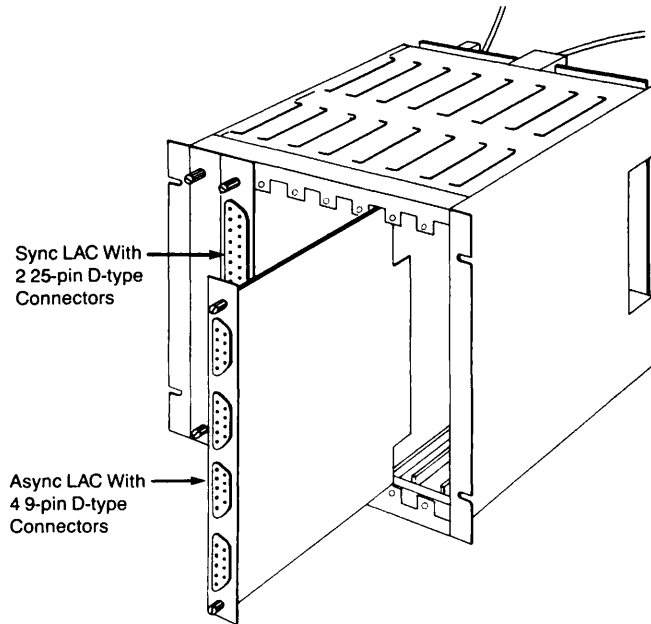
FIGURE 1-5. An 8-slot ICS3 Card Cage

Caution

Do not touch the LACs in the card cage or tamper with the cables when the power is on. Damage may result.

Follow this basic procedure for installing LACs:

1. Verify that the power is off.

2. Locate the card cage.

3. Remove the card-support bar that spans the front of the card cage.

4. Locate any vacant card slot to the right of the buffer card in the backplane.

5. Hold the new card vertically with its component side facing left and its edge connectors pointing toward the intended backplane slot.

6. Align the card with the upper and lower card guides and insert it into the slot until the edge connectors protrude through the backplane and the ejector tabs are engaged.

7. Simultaneously close the upper and lower ejector tabs located on the front of the card. Press the tabs until the card is secure.

8. When all cards are installed, replace the support bar.

9. Restore power and boot the system.

10. If you are using the ICS CARDS directive, change it to agree with the new configuration and initialize the LAC by rebooting the host computer system, when scheduling allows.

LHC Controller Characteristics

The LAN Host Controller (LHC) is a microprocessor-based communication controller that supports a IEEE 802.3 local area network. IEEE 802.3 networks use the Carrier Sense Multiple Access Collision Detector (CSMA/CD) access control method to transmit data at 10 MbPS over a standard media in a bus topology. Figure 1-6 shows an LHC board.

The LHC configuration directive assigns a 2-digit PRIMOS octal device address to a LHC. Valid device addresses for an LTS are 10_8, 11_8, 15_8, 16_8, 17_8, 32_8, 35_8, 36_8, 37_8, 50_8, 51_8, 52_8, 53_8, 54_8, and 56_8.

FIGURE 1-6. Lan Host Controller Board

Each LHC is assigned a unique hexadecimal address by the manufacturer. This address is stamped on a metal tag attached to the device. It is used during NTS configuration.
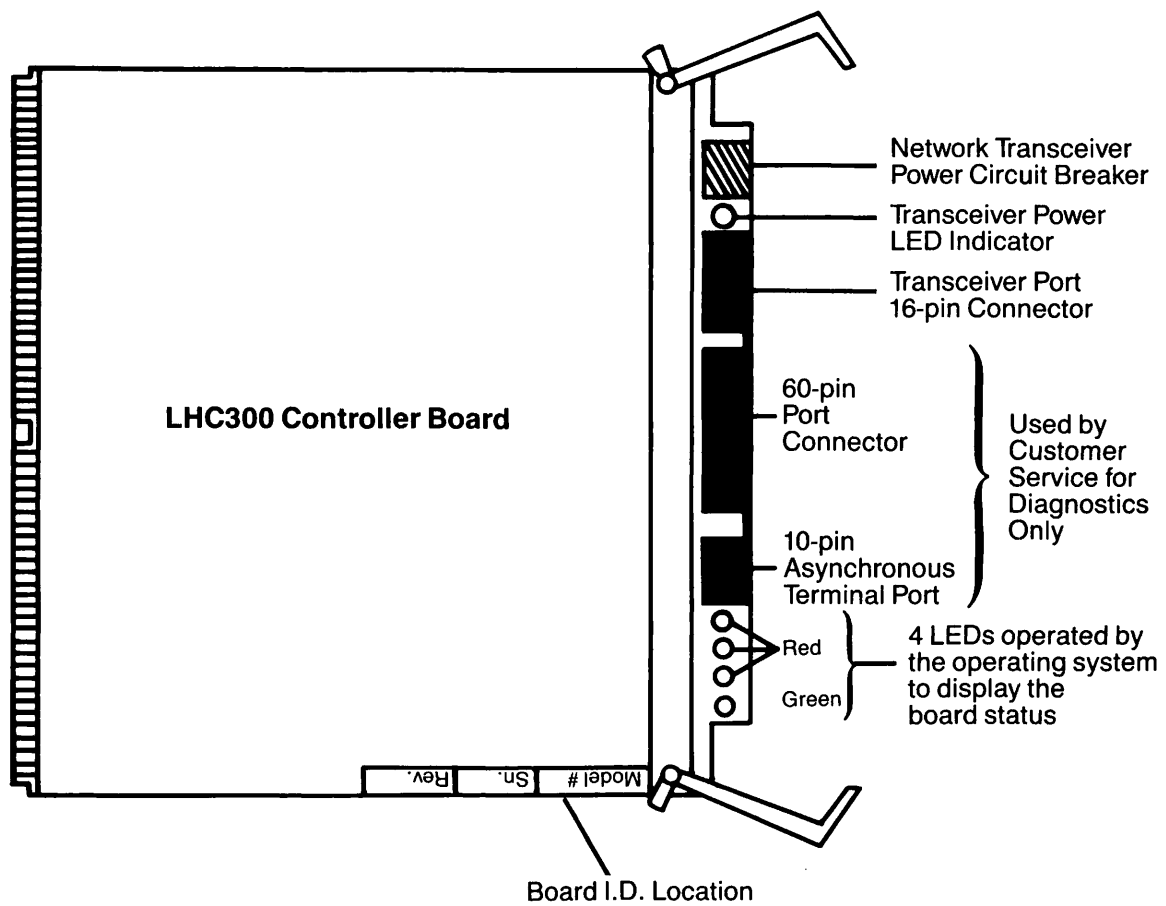
You can downline load and upline dump LHC controllers interactively or at cold start with the COMM_CONTROLLER command. For more information on this procedure, see Chapter 2, Operating Intelligent Controllers.

An LHC controller can support four different protocol sets

- PRIMENET

- NTS

- PRIMENET and NTS concurrently

- TCP/IP

At PRIMOS Rev. 21.0, you can install LHC controllers in 50 series machines. However, the following configuration limits apply:

- A maximum of 6 LHC controllers can be installed on one system provided you have available slots in the backplane and sufficient amperage.

- A maximum of 2 LHCs can be configured for PRIMENET on one system.

- A maximum of 4 LHCs can be configured for NTS on one system.

- Each LHC requires a transceiver or a port on a multiport transceiver to connect to the network.

- Each LHC can not be located more than 50 meters from the network cable and transceiver.

- A multiport receiver is required when more than 3 LHCs are installed in one host.

For more information on LHCs and NTS, see the *NTS Planning and Configuration Guide.*

LTS Characteristics

A LAN Terminal Server (LTS) is a standalone device that permits 8 asynchronous RS-232-C devices to connect to an IEEE 802.3 local area network with NTS and PRIMENET software. Terminals connected to the LTS can communicate with any host on the LAN.

Each LTS is assigned a unique hexadecimal device address by the manufacturer, which is located on a metal tag attached to the LTS. This address is used when the LTS is configured, downline loaded, and upline dumped.

A host is a 50 Series machine or another computer system connected to the LAN. A node is another host or an LTS. An LTS requires at least one Prime host on the LAN to downline load its operating software. You can use the COMM_CONTROLLER command to

downline load and upline dump an LTS. See Chapter 2, Operating Intelligent Controllers, for more information on the COMM_CONTROLLER command. Once an LTS is booted, however, it owes no affiliation to a host and is completely independent.

If the network allows unconfigured nodes, you can install an LTS by plugging it into the transceiver. NTS can configure the LTS without interrupting the network and halting traffic on the LAN. Otherwise follow the procedure given in the *NTS Planning and Configuration Guide*. For more information on LTSs, NTS, and a discussion of configured and unconfigured nodes, see the *NTS Planning and Configuration Guide*.

The back panel of an LTS is shown in Figure 1-7. Each LTS has eight standard RS-232-C 25-pin connector ports numbered 0 through 7. The XCVR port connects the LTS to the transceiver, which is attached to the LAN. Each LTS also has a cascade port that allows a maximum of 4 LTSs to connect to a single tranceiver as shown in Figure 1-8. This allows a maximum of 32 asynchronous devices to use a single network tranceiver connection.

The internal memory of an LTS is preset at the factory to detect the baud rate for input devices connected to the RS-232-C ports. It recognizes the following baud rates: 110, 300, 600, 1200, 2400, 4800, 7200, 9600, and 19200.
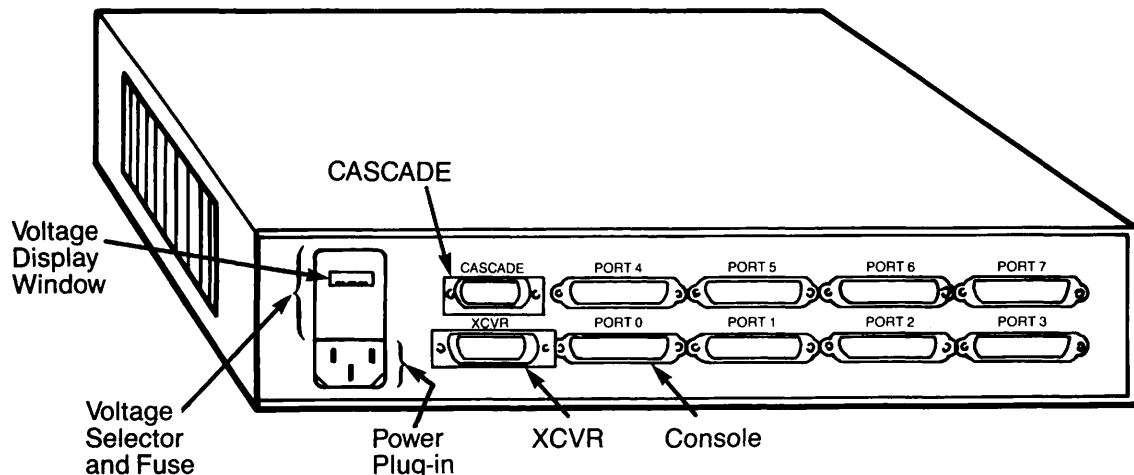
FIGURE 1-7. The Rear Panel of a LAN Terminal Server

You can activate ASD at any properly set terminal attached to an LTS port by keying CR several times. The LTS responds with a reply and a prompt. At this point you can change the baud rate and parity of the port with the PORT or SET commands. You can store these values and other user-defined line parameters in non-volatile memory. The system maintains these parameters when the LTS is rebooted or powered off, until you change them. For more information on LTS commands, see the *LTS User's Guide.*

Network
Transceiver

Network
Cable

LTS300s

Transceiver
Cable

Interconnection
Cable
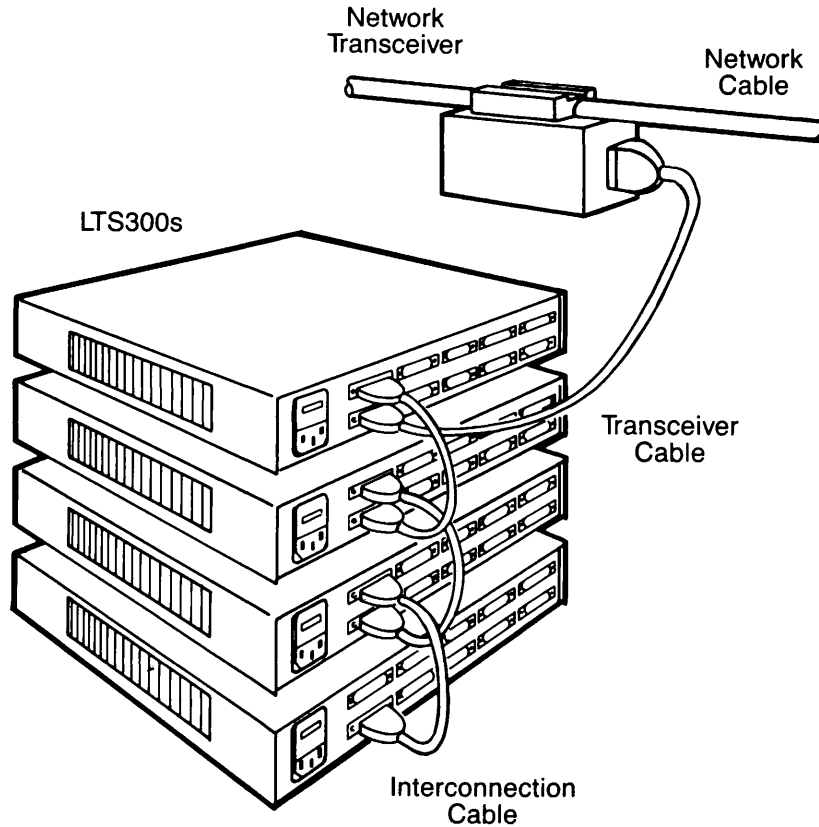
FIGURE 1-8. An LTS Cascade Configuration

OPERATING INTELLIGENT CONTROLLERS

Communication controllers are devices that direct the transmission of data between the CPU and either locally connected devices or a network. Some communication controllers are managed by a program executing in the host. Intelligent controllers, such as ICS and LHC controllers, are managed by a program executing in the microprocessor on the controller board. These programs are stored in the host and transferred to the on-board microprocessor with either configuration directives or interactive commands.

This chapter introduces the COMM_CONTROLLER command, an interactive method to initialize, downline load, upline dump, and shut down your communication controllers.

Prior to Rev. 21.0, the configuration directives SYNC ON, SYNC DSC, SYNC CNTRLR, and SYNC SYNCnn enabled and loaded ICS controllers at cold start. You can still use SYNC directives in your CONFIG file; but you can also use the COMM_CONTROLLER command interactively while your system is running. SYNC directives are explained in detail in Chapter 3, Useful Configuration Directives.

The COMM_CONTROLLER Command

The COMM_CONTROLLER command reboots, downline loads, upline dumps, or stops an intelligent controller while the system is running without affecting any other controllers. You can issue the command interactively for a single controller or for a number of controllers, provided that all the affected controllers are the same type and are attached to the same system.

Access to this command is restricted to System and Network Administrators working at the supervisor terminal. DSM logs unauthorized attempts to use the COMM_CONTROLLER command, if DSM is running when the COMM_CONTROLLER command is issued. You can put COMM_CONTROLLER commands in your PRIMOS.COMI file; place them after the START_DSM command and before the START_NET and START_NTS commands. When you use COMM_CONTROLLER commands in your PRIMOS.COMI file or in CPL programs, use the NO_QUERY option to suppress prompts.
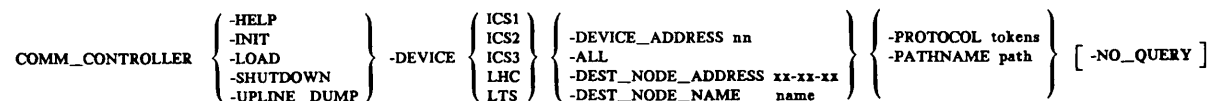
COMM_CONTROLLER Subcommands

The main COMM_CONTROLLER command is actually a command interpreter. The main command parses the command line, detects input errors, and invokes one of the five subcommands. The subcommand spawns a specialized phantom process, known as a server, to actually perform the desired operation. The five main COMM_CONTROLLER subcommands are listed below with a brief description of their functionality.

Subcommand	*Description*
COMM_CONTROLLER -HELP	Displays the command line format, a brief description of the five subcommands, and a list of options.
COMM_CONTROLLER -INIT	Initializes ICS1, ICS2, and ICS3 controllers.
COMM_CONTROLLER -LOAD	Reboots the controller and downline loads a specific file or protocol combination from the host to the controller. Valid for ICS1, ICS2, ICS3, LHC, and LTS controllers.
COMM_CONTROLLER -SHUTDOWN	Shuts down ICS1, ICS2, and ICS3 controllers.
COMM_CONTROLLER -UPLINE_DUMP	Writes the contents of an LHC or LTS microprocessor's load file to a file in the host for analysis or storage.

The COMM_CONTROLLER subcommands support several addressing schemes, protocol tokens, and load file options for a variety of controllers. Figure 2-1 illustrates the complete command line format. Not all subcommands and options are available for every controller. Each subcommand is described in this chapter in a separate section, which lists any exceptions and device-specific instructions.

```
                      ( -HELP        )           ( ICS1 )  ( -DEVICE_ADDRESS  nn           )  ( -PROTOCOL tokens )
                      | -INIT        |           | ICS2 |  | -ALL                          |  | -PATHNAME path   |
COMM_CONTROLLER       < -LOAD        >  -DEVICE  < ICS3 >  < -DEST_NODE_ADDRESS  xx-xx-xx   >  <                  >  [ -NO_QUERY ]
                      | -SHUTDOWN    |           | LHC  |  | -DEST_NODE_NAME      name       |  (                  )
                      ( -UPLINE_DUMP )           ( LTS  )  ( -DEST_NODE_NAME      name       )
```

FIGURE 2-1. The Complete COMM_CONTROLLER Command Line Format

Creating Abbreviations

When a frequently used command is especially long or complicated, an abbreviation can make it easier to remember, save time, and reduce input errors. If abbreviations are allowed on your system, you can use the ABBREV command to create abbreviations for the COMM_CONTROLLER command. For example, the following command line creates the abbreviation QUIET to stop all controllers on the system for the device type specified.

```
ABBREV -ADD_COMMAND QUIET COMM_CONTROLLER -SHUT_DOWN -DEVICE %1% -ALL -NO_QUERY
```

When you issue the personal abbreviation QUIET and substitute a valid device type for the variable %1%, PRIMOS expands the abbreviation and executes the full command line. For example,

```
QUIET ICS2
```

is equivalent to

```
COMM_CONTROLLER -SHUT_DOWN -DEVICE ICS2 -ALL -NO_QUERY
```

Note
Do not use personal abbreviations in your PRIMOS.COMI file.

Any abbreviations that you create at the supervisor terminal for the COMM_CONTROLLER command are available to any user logged on as SYSTEM at the supervisor terminal. To avoid possible conflicts, do not choose an existing PRIMOS command or its abbreviation for a user-defined abbreviation. See the *PRIMOS Commands Reference Guide* for an extensive description of user-defined abbreviations and the ABBREV command.

The COMM_CONTROLLER -HELP Subcommand

The COMM_CONTROLLER -HELP subcommand displays detailed information on all COMM_CONTROLLER subcommands and arguments. When you choose the -HELP subcommand, all other subcommands and options are ignored.

The format of this subcommand is

COMM_CONTROLLER $\left\{ \begin{array}{l} \text{-HELP} \\ \text{-H} \end{array} \right\}$

When you issue the COMM_CONTROLLER command without specifying a subcommand, the command interpreter displays the help file.

The COMM_CONTROLLER -INIT Subcommand

The COMM_CONTROLLER -INIT Subcommand initializes ICS controllers. This command shuts down the specified controller and activates the self-verification test programs resident in the on-board microprocessor. When testing is complete, the available memory size is returned to the CPU and the controller is ready to be started. Any error messages are displayed at the supervisor terminal and, if desired, logged by DSM. Do not specify the -INIT subcommand with the -LOAD subcommand.

This subcommand is not implemented for LHC and LTS controllers.

The format of this subcommand is

$$\text{COMM\_CONTROLLER -INIT -DEVICE} \quad \left\{ \begin{array}{l} \text{-ICS1} \\ \text{-ICS2} \\ \text{-ICS3} \end{array} \right\} \quad \left\{ \begin{array}{l} \text{-DEVICE\_ADDRESS nn} \\ \text{-ALL} \end{array} \right\} \quad [\text{ -NO\_QUERY }]$$

This subcommand has the following options:

Option	*Description*
$\left\{ \begin{array}{l} \text{-DEVICE type} \\ \text{-DEV} \end{array} \right\}$	Specifies the type of ICS controller. Valid device types are ICS1, ICS2, or ICS3. Use -DEVICE with either the -DEVICE_ADDRESS or the -ALL option.
$\left\{ \begin{array}{l} \text{-DEVICE\_ADDRESS nn} \\ \text{-DA} \end{array} \right\}$	Specifies an individual ICS controller's device address. Obtain this two-digit octal number with the STATUS COMMS or LIST_COMM_CONTROLLERS commands. Valid ICS device addresses for Rev. 21.0 are 10_8, 11_8, 36_8, and 37_8. Commands containing invalid addresses are rejected and generate an error message that lists all controllers and their addresses.
-ALL	Initializes every ICS controller of the *type* specified in the -DEVICE option on the system.
$\left\{ \begin{array}{l} \text{-NO\_QUERY} \\ \text{-NQ} \end{array} \right\}$	Suppresses the Continue? prompt. Use the -*NQ* option whenever you include this command in a CPL program or in your PRIMOS.COMI file.

Examples of the COMM_CONTROLLER -INIT Subcommand

In the following examples two ICS2 controllers reside on a single system. The first COMM_CONTROLLER command initializes the ICS2 controller at device address 10_8. The second COMM_CONTROLLER command initializes both ICS2 controllers.

Initializing One Controller:

```
OK, COMM_CONTROLLER -INIT -DEVICE ICS2 -DEVICE_ADDRESS 10

[COMM_CONTROLLER Rev. 21.0 Copyright (c) 1987, Prime Computer, Inc.]

ICS2_SERVER (user 185) logged in Friday, 27 Mar 87 10:53:20.
ICS2 controller(s) 10 currently active. Continue <Y or N>? Y

ICS2 controller shutdown in progress...

ICS2 prom self-verify diagnostics in progress...

ICS2 initialization operation results:
     ICS2 at address 10: SUCCESSFUL
OK,
ICS2_SERVER (user 185) logged out Friday, 27 Mar 87 10:53:44.
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.

Phantom 185: Normal logout at 10:53
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.
```

Initializing Multiple Controllers:

```
OK, COMM_CONTROLLER -INIT -DEVICE ICS2 -ALL

[COMM_CONTROLLER Rev. 21.0 Copyright (c) 1987, Prime Computer, Inc.]

ICS2_SERVER (user 185) logged in Friday, 27 Mar 87 10:53:20.
ICS2 controller(s) 10 11 currently active. Continue <Y or N>? Y

ICS2 controller shutdown in progress...

ICS2 prom self-verify diagnostics in progress...

ICS2 initialization operation results:
     ICS2 at address 10: SUCCESSFUL
     ICS2 at address 11: SUCCESSFUL
OK,
ICS2_SERVER (user 185) logged out Friday, 27 Mar 87 10:53:44.
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.

Phantom 185: Normal logout at 10:53
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.
```

The COMM_CONTROLLER -LOAD Subcommand

The COMM_CONTROLLER -LOAD subcommand executes an automatic shutdown and reinitialization cycle followed by a downline load of a specific file or protocol combination to the specified controller(s). You must give the LHC_DLL_SERVER and the LTS_DLL_SERVER all access rights to the directory DOWN_LINE_LOAD* and any other directory containing downline load files. You must have .NETWORK_MGT access to downline load an LTS controller. Any error messages are displayed at the supervisor terminal and, if desired, are logged by DSM.

Note

The START_NET and START_NTS commands do not downline load LHC or LTS controllers. Both devices must be downline loaded explicitly with the COMM_CONTROLLER command.

The -LOAD subcommand completely reboots a controller; do not specify the -INIT subcommand at the same time.

The complete format of this subcommand is

$$\text{COMM\_CONTROLLER -LOAD -DEVICE} \left\{ \begin{matrix} \text{ICS1} \\ \text{ICS2} \\ \text{ICS3} \\ \text{LHC} \\ \text{LTS} \end{matrix} \right\} \left\{ \begin{matrix} \text{-DEVICE\_ADDRESS nn} \\ \text{-ALL} \\ \text{-DEST\_NODE\_ADDRESS xx-xx-xx} \\ \text{-DEST\_NODE\_NAME name} \end{matrix} \right\} \left\{ \begin{matrix} \text{-PROTOCOL tokens} \\ \text{-PATHNAME path} \end{matrix} \right\} \left[\text{ -NO\_QUERY } \right]$$

This subcommand has the following options:

Option	*Description*
{ **-DEVICE type** **-DEV** }	Specifies the type of intelligent controller. Valid device types are ICS1, ICS2, ICS3, LHC, or LTS. Use with either the -DA or the -ALL option.
{ **-DEVICE_ADDRESS nn** **-DA** }	Specifies an individual ICS or LHC controller's device address. This two-digit octal number is obtained with the STATUS COMMS or LIST_COMM_CONTROLLERS commands. Valid ICS device addresses for Rev. 21.0 are 10_8, 11_8, 36_8, and 37_8. Valid LHC addresses at Rev. 21.0 are 10_8, 11_8, 15_8, 16_8, 17_8, 32_8, 35_8, 36_8, 37_8, 50_8, 51_8, 52_8, 53_8, 54_8, and 56_8. Commands containing invalid addresses are rejected and generate an error message that lists all available controllers and their addresses.
-ALL	Loads every controller of the *type* specified in the -DEVICE

option on the system with identical protocol token combinations or the same downline load file.

{ **-DEST_NODE_ADDRESS**
  -DNA }

Specifies the unique 12-digit hexadecimal network address of an LTS. This address is assigned by the manufacturer and is located on a metal tag attached to the LTS. All Prime LTS addresses begin with the three-digit string 08-00-2F-. When you enter only the last six digits of the device address, the COMM_CONTROLLER command adds the Prime LTS prefix automatically. Separate every two characters with a hyphen. You can also use the LIST_LTS_STATUS command to obtain the device address. Do not specify the -DNA option with the -DNN option. This option applies to LTS controller only.

{ **-DEST_NODE_NAME**
  -DNN }

Specifies the unique node name of an LTS. This name is assigned by CONFIG_NTS when the LTS is configured. Use the LIST_LTS_STATUS command to obtain the node name. Do not specify the -DNN option with the -DNA option. This option applies to LTS controllers only.

{ **-PROTOCOL tokens**
  -PR }

Specifies the protocol token combinations used to select a downline load file. Protocol tokens define the synchronous and asynchronous parameters available for lines connected to the controller.

You can specify tokens in any order provided that they are joined by an underscore. For example, ASYNC_BSCRJE is the same as BSCRJE_ASYNC. This option is functionally equivalent to the SYNC CNTRLR configuration directive discussed in Chapter 3, Useful Configuration Directives. The following protocol tokens are currently available:

Type	*Tokens*
ICS	HDLC, SDLC, ASYNC, BSCX25, or BSCRJE
LHC	LLCX25, NTS, or TCP_TEL

Note

Do not specify NTS and TCP_TEL for an LHC controller.

{ **-PATHNAME path**
  -PN }

Transfers a file containing executable microcode from storage to the controller. Do not use the -PATHNAME option with the -PROTOCOL option. *path* must be an existing PRIMOS filename no longer than 128 characters. If you are attached to the source

directory, you can specify just the filename. The DOWN_LINE_LOAD* directory contains the following pre-built object code files for each device type:

```
DOWN_LINE_LOAD*>ICS1.DL
DOWN_LINE_LOAD*>ICS2_01.DL (ASYNC only)
DOWN_LINE_LOAD*>ICS3_01.DL (ASYNC only)
DOWN_LINE_LOAD*>LHC.DL
DOWN_LINE_LOAD*>LTS.DL
```

-NO_QUERY **-NQ**	Suppresses the Continue? prompt. Use the *-NQ* option whenever you include this command in a CPL program or in your PRIMOS.COMI file.

Examples of the COMM_CONTROLLER -LOAD Subcommand

Loading One LTS: The following command line in a PRIMOS.COMI file downline loads the LTS named ACCTG.DEPT.

```
COMM_CONTROLLER -LOAD -DEV LTS -DNN ACCTG.DEPT -PN DOWN_LINE_LOAD*>LTS.DL -NQ
```

Loading an LHC Controller: LHC controllers that are running NTS and PRIMENET/LAN300 concurrently require two protocol tokens: NTS and LLCX25. Use the -PROTOCOL option to load this combination. Connect the tokens with an underscore as shown in the following example.

```
OK, COMM_CONTROLLER -LOAD -DEVICE LHC  -DEVICE_ADDRESS 37 -PROTOCOL LLCX25_NTS

[COMM_CONTROLLER Rev. 21.0 Copyright (c) 1986, Prime Computer, Inc.]
LHC controller(s) 37 currently active. Continue <Y or N>? Y

LHC prom self-verify diagnostics in progress...

LHC downline load in progress...

LHC downline load operation results:
     LHC at address 37: SUCCESSFUL
OK,
Phantom 180: Normal logout at 13:06
Time used: 00h 00m connect, 00m 07s CPU, 00m 06s I/O.
```

Loading Several Controllers: You can load several controllers with one command line provided that the controllers

- Are the same type

- Are located in the same backplane

- Use the same downline load file or protocol combination

The following command line loads every ICS3 controller on this system with three protocols: ASYNC, HDLC, and BSCX25.

```
OK, COMM_CONTROLLER -LOAD -DEVICE ICS3 -ALL -PROTOCOL ASYNC_HDLC_BSCX25

[COMM_CONTROLLER Rev. 21.0 Copyright (c) 1987, Prime Computer, Inc.]

ICS3_SERVER (user 196) logged in Friday, 27 Mar 87 10:55:12.
Continue <Y or N>? Y

ICS3 controller shutdown in progress...

ICS3 prom self-verify diagnostics in progress...

ICS3 downline load in progress...

ICS3 run-time self-verify diagnostics in progress...

ICS3 controller startup in progress...

ICS3 downline load operation results:
     ICS3 at address 10: SUCCESSFUL
     ICS3 at address 11: SUCCESSFUL
OK,
ICS3_SERVER (user 196) logged out Friday, 27 Mar 87 10:55:32.
Time used: 00h 00m connect, 00m 03s CPU, 00m 01s I/O.

Phantom 196: Normal logout at 10:55
Time used: 00h 00m connect, 00m 03s CPU, 00m 01s I/O.
```

The COMM_CONTROLLER -SHUTDOWN Subcommand

The COMM_CONTROLLER -SHUTDOWN subcommand suspends the specified ICS controller(s). This subcommand breaks all logical connections between the controller and peripheral devices. Any error messages are displayed at the supervisor terminal and, if desired, logged by DSM.

This subcommand is not implemented for LHC and LTS controllers.

Caution

Do not shut down controllers that support complete SNA, PRIMENET/X.25, or RJE subsystems. These applications do not recover their lines automatically and must be restarted.

The format of this subcommand is

$$
\text{COMM\_CONTROLLER} \left\{ \begin{matrix} \text{-SHUT\_DOWN} \\ \text{-SH} \end{matrix} \right\} \quad \text{-DEVICE} \left\{ \begin{matrix} \text{ICS1} \\ \text{ICS2} \\ \text{ICS3} \end{matrix} \right\} \left\{ \begin{matrix} \text{-DEVICE\_ADDRESS nn} \\ \text{-ALL} \end{matrix} \right\} \quad [\ \text{-NO\_QUERY}\ ]
$$

This subcommand has the following options:

Option	*Description*
-DEVICE type -DEV	Specifies the type of ICS controller. Valid device types are ICS1, ICS2, or ICS3. Use with either the -DEVICE_ADDRESS or the -ALL option.
-DEVICE_ADDRESS nn -DA	Specifies an individual ICS controller's device address. Obtain this two-digit octal number with the STATUS COMMS or LIST_COMM_CONTROLLERS commands. Valid ICS device addresses for Rev. 21.0 are 10_8, 11_8, 36_8, and 37_8. Commands containing invalid addresses are rejected and generate an error message that lists all available controllers and their addresses.
-ALL	Shuts down every ICS controller of the *type* specified in the -DEVICE option on the system.
-NO_QUERY -NQ	Suppresses the Continue? prompt. Use the *-NQ* option whenever you include this command in a CPL program or in your PRIMOS.COMI file.

Examples of the COMM_CONTROLLER -SHUTDOWN Subcommand

The first command line example shuts down an individual ICS2 controller with the device address 11_8; the second command line example shuts down every ICS2 controller in the backplane.

Suspending One Controller:

```
OK,  COMM_CONTROLLER -SHUTDOWN -DEVICE ICS2 -DEVICE_ADDRESS 10

[COMM_CONTROLLER Rev. 21.0 Copyright (c) 1987, Prime Computer, Inc.]

ICS2_SERVER (user 197) logged in Friday, 27 Mar 87 10:54:24.
ICS2 controller(s) 11 currently active. Continue <Y or N>? Y

ICS2 controller shutdown in progress...

ICS2 shutdown operation results:
     ICS2 at address 11: SUCCESSFUL
OK,
ICS2_SERVER (user 197) logged out Friday, 27 Mar 87 10:54:44.
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.

Phantom 197: Normal logout at 10:54
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.
```

Suspending Several Controllers:

```
OK, COMM_CONTROLLER -SHUTDOWN -DEVICE ICS2 -ALL
[COMM_CONTROLLER Rev. 21.0 Copyright (c) 1987, Prime Computer, Inc.]

ICS2_SERVER (user 197) logged in Friday, 27 Mar 87 10:54:24.
ICS2 controller(s) 11 36 currently active. Continue <Y or N>? Y

ICS2 controller shutdown in progress...

ICS2 shutdown operation results:
     ICS2 at address 11: SUCCESSFUL
     ICS2 at address 36: SUCCESSFUL
OK,
ICS2_SERVER (user 197) logged out Friday, 27 Mar 87 10:54:44.
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.

Phantom 197: Normal logout at 10:54
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.
```

The COMM_CONTROLLER -UPLINE_DUMP Subcommand

The COMM_CONTROLLER -UPLINE_DUMP subcommand commands an LHC or LTS controller to write the contents of predetermined sections of its memory into the specified file in the host for subsequent analysis. Only one controller can upline dump data to the host at a time. Any error messages are displayed at the supervisor terminal and, if desired, logged by DSM.

Caution

Before the COMM_CONTROLLER -UPLINE_DUMP subcommand can open a file and write data, the LHC_ULD_SERVER and the LTS_ULD_SERVER must have ALL access rights to the directory containing the file specified by the -PATHNAME option or to UPLINE_DUMP*>LAN300 for the default case.

The format of this subcommand is

COMM_CONTROLLER { -UPLINE_DUMP / -ULD } -DEVICE { LHC / LTS } { -DEVICE_ADDRESS nn / -DEST_NODE_ADDRESS xx-xx-xx / -DEST_NODE_NAME name } -PATHNAME path [-NO_QUERY]

This subcommand has the following options:

Option   *Description*

-DEVICE type ⎫
-DEV   ⎭   Specifies the type of intelligent controller. Valid device types are
                      LHC or LTS.

-DEVICE_ADDRESS nn ⎫
-DA   ⎭

                      Specifies an individual LHC controller's device address. Obtain
                      this two-digit octal number with the STATUS COMMS or
                      LIST_COMMS_CONTROLLER commands.   Valid LHC addresses
                      at Rev. 21.0 are 10_8, 11_8, 15_8, 16_8, 17_8, 32_8, 35_8, 36_8, 37_8, 50_8,
                      51_8, 52_8, 53_8, 54_8, and 56_8.   Commands containing invalid
                      addresses are rejected and generate an error message that lists all
                      available controllers and their addresses.

-DEST_NODE_ADDRESS ⎫
-DNA   ⎭

                      Specifies the unique 12-digit hexadecimal network address of an
                      LTS. This address is assigned by the manufacturer and is located
                      on a metal tag attached to the LTS. All Prime LTS addresses
                      begin with the three-digit string 08-00-2F-. When you enter only
                      the last six digits of the device address, the
                      COMMS_CONTROLLER command adds the Prime LTS prefix
                      automatically. Separate every two characters with a hyphen.
                      You can also use the LIST_LTS_STATUS command to obtain
                      the device address.   Do not specify the -DNA option with the
                      -DNN option.   This option applies to LTS controllers only.

-DEST_NODE_NAME ⎫
-DNN   ⎭
                      Specifies the unique node name of an LTS.   This name is
                      assigned with CONFIG_NTS when the LTS is configured. Use
                      the LIST_LTS_STATUS command to obtain the node name.   Do
                      not specify the -DNN option with the -DNA option.   This option
                      applies to LTS controllers only.

-PATHNAME path ⎫
-PN   ⎭   Opens the file specified by *path*, where *path* is a PRIMOS
                      pathname of not more than 128 characters.   If you specify just
                      a filename, the file is opened in your currently attached
                      directory.

                      When a pathname is omitted, the server retrieves the current
                      memory image from the device and stores it in a file in the
                      directory UP_LINE_DUMP*>LAN300. The format for default
                      filenames is *device_host-address_date.time*.

                      The convention for upline dump filenames in the
                      DOWN_LINE_LOAD* directory is MODEL_NUMBER.ULD. For
                      example, the first ICS3 controller's upline dump filename is
                      DOWN_LINE_LOAD*>ICS3_01.ULD.

⎰ **-NO_QUERY** ⎱   Suppresses the Continue? prompt. Use the *-NQ* option whenever
⎱ **-NQ**   ⎰   you include this command in a CPL program or in your
                                   PRIMOS.COMI file.

Examples of the COMM_CONTROLLER -ULD Subcommand

Using the DEVICE_ADDRESS Option: In the following example, the command line transfers the contents of an LHC controller's on-board microprocessor to separate files in the host.

```
OK, COMM_CONTROLLER -ULD -DEV LHC  -DA 50 -PN UP_LINE_DUMP*>LAN300>LHC_02.ULD

[COMM_CONTROLLER Rev. 21.0 Copyright (c) 1987, Prime Computer, Inc.]

LHC_ULD_SERVER (user 197) logged in Friday, 27 Mar 87 10:54:24.
LHC controller(s) 50 currently active. Continue <Y or N>? Y

LHC controller upline dump in progress...

LHC upline dump operation results:
    LHC at address 50: SUCCESSFUL
OK,
LHC_ULD_SERVER (user 197) logged out Friday, 27 Mar 87 10:54:44.
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.

Phantom 197: Normal logout at 10:54
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.
```

Using the DESTINATION_NODE_NAME Option: The COMM_CONTROLLER command can upline dump an LTS by node name as shown in the example below.

```
OK, COMM_CONTROLLER -ULD -DEV LTS -DNN SALES.DEPT -PN DOWN_LINE_LOAD*>LTS_03.ULD

[COMM_CONTROLLER Rev. 21.0 Copyright (c) 1987, Prime Computer, Inc.]

LTS_ULD_SERVER (user 197) logged in Friday, 27 Mar 87 10:54:24.
LTS controller(s) SALES.DEPT currently active. Continue <Y or N>? Y

LTS controller upline dump in progress...

LTS upline dump operation results:
    LTS at DNN SALES.DEPT: SUCCESSFUL

OK,
LTS_ULD_SERVER (user 197) logged out Friday, 27 Mar 87 10:54:44.
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.

Phantom 197: Normal logout at 10:54
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.
```

Using the DESTINATION_NODE_ADDRESS Option: The COMM_CONTROLLER command can upline dump an LTS by network address as shown in the example below.

```
OK, COMM_CONTROLLER -ULD -DEV LTS -DNA F8-00-01 -PN DOWN_LINE_LOAD*>LTS_03.ULD

[COMM_CONTROLLER Rev. 21.0 Copyright (c) 1987, Prime Computer, Inc.]

LTS_ULD_SERVER (user 197) logged in Friday, 27 Mar 87 10:54:24.
LTS controller(s) 08-00-2F-F8-00-01 currently active. Continue <Y or N>? Y

LTS controller upline dump in progress...

LTS upline dump operation results:
    LTS at DNA 08-00-2F-F8-00-01: SUCCESSFUL

OK,
LTS_ULD_SERVER (user 197) logged out Friday, 27 Mar 87 10:54:44.
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.

Phantom 197: Normal logout at 10:54
Time used: 00h 00m connect, 00m 02s CPU, 00m 01s I/O.
```

COMM_CONTROLLER Command Error Messages

The COMM_CONTROLLER command interpreter displays the copyright notice on the screen while it verifies the following:

- Authorized user at supervisor terminal

- Adequate access rights for servers

- Correct command line syntax

- Valid options, addresses, and names

When an error is detected, the command request is rejected and an error message is displayed. Normally, all error messages are also logged by DSM. The error messages listed below are the most common. The *ICS User's Guide* contains a complete list of the ICS interface error messages and boot server logic error messages.

Error: ALL option invalid for LTS.
    LTS controllers must be loaded and upline dumped separately.

Error: DEST_NODE_ADDRESS contains incorrect number of hexadecimal digits.
    A valid LTS destination node address is a string of six or twelve hexadecimal digits separated by hypens.

Error: DEST_NODE_ADDRESS has too many contiguous hexadecimal digits.
    A valid LTS destination node address is a string of six or twelve hexadecimal digits separated by hyphens.

Error: DEST_NODE_ADDRESS not specified.
    The DEST_NODE_ADDRESS option was specified without the hexadecimal address string.

`Error: DEST_NODE_ADDRESS option is invalid for the <type> controller.`
The DEST_NODE_ADDRESS option was specified for an ICS controller. Only LTS and LHC controllers can use this option.

`Error: DEST_NODE_ADDRESS separator <-> arranged incorrectly.`
There is a misplaced hyphen in the hexadecimal address string.

`Error: DEST_NODE_NAME not specified.`
The DEST_NODE_NAME option was specified without a valid LTS name.

`Error: DEST_NODE_NAME too long.`
The LTS node name supplied in the command line exceeds the maximum length of 16 characters.

`Error: DEVICE <type> is invalid.`
The controller *type* specified must be one of the following: ICS1, ICS2, ICS3, LHC, or LTS.

`Error: DEVICE not specified.`
The DEVICE option was specified without supplying a valid controller type.

`Error: DEVICE option must be specified.`
The DEVICE option was omitted from the command line.

`Error: DEVICE_ADDRESS <nn> is not octal.`
The device address for a communication controller must be a two-digit octal number.

`Error: DEVICE_ADDRESS has more than two characters.`
The device address for a communication controller must be a two-digit octal number.

`Error: DEVICE_ADDRESS not specified.`
The DEVICE_ADDRESS option was specified without a valid address.

`Error: DEVICE_ADDRESS option invalid for LTS.`
Use either the DNA or the DNN options to identify an LTS; the DEVICE_ADDRESS option is only for locally-attached controllers.

`Error: DNN option is invalid for the <type> controller.`
The DNN option is only valid for LTS controllers.

`Error: INIT option invalid for the <type> controller.`
The INIT option applies only to ICS controllers.

`Error: PATHNAME <path> is invalid.`
The pathname must be an existing PRIMOS pathname.

`Error: PATHNAME not specified.`
The PATHNAME option was specified without supplying a valid PRIMOS pathname.

`Error: PATHNAME option can only be specified with LOAD or ULD.`
The PATHNAME option was supplied with either the INIT or SHUTDOWN options.

Error: PROTOCOL option can only be specified with load.
The PROTOCOL option was supplied with either the INIT, SHUTDOWN, or UPLINE__DUMP option.

Error: PROTOCOL option is invalid for the <type> controllers.
LTS and ICS1 controllers do not support the PROTOCOL option.

Error: PROTOCOL separator <_> arranged incorrectly.
Valid protocol token combinations are connected by an underscore.

Error: PROTOCOL token <token> is duplicated.
A protocol token was supplied more than once in the command line.

Error: PROTOCOL token <token> is invalid.
The only valid protocols for Rev. 21.0 are LLCX25, NTS, TCP, TEL, ASYNC, SYNC, HDLC, BSCRJE, and BSCX25.

Error: PROTOCOL token is not specified.
The PROTOCOL option was specified without a valid token combination.

Error: SHUTDOWN option invalid for the <type> controller.
The SHUTDOWN option was specified for either an LHC or LTS controller.

Error: UPLINE_DUMP option invalid for the <type> controller.
The UPLINE__DUMP option was specified for an ICS controller.

Error: cannot specify DEVICE_ADDRESS and ALL together.
Do not specify both options on one command line.

Error: cannot specify DEST_NODE_NAME and DEST_NODE_ADDRESS together.
Do not specify both options on one command line.

Error: cannot specify PATHNAME and PROTOCOL together.
Do not specify both options on one command line.

Error: cannot upline dump more than ONE device.
More than one controller was specified on the command line.

Error: either DEST_NODE_NAME or DEST_NODE_ADDRESS option must be specified.
The device type LTS was specified without further identification.

Error: either the DEVICE_ADDRESS or ALL option must be specified.
Either an LHC or an ICS controller was specified without further identification.

Error: invalid PATHNAME specified.
PATHNAME must be an existing PRIMOS pathname no longer than 128 characters.

Error: invalid response to a question.
The only valid answers to the Continue? prompt are Y or N.

Error: no main option specified (select LOAD, ULD, SH, INIT or HELP).
The COMM__CONTROLLER command was issued without specifying an operation.

```
Error:  unknown option <option>.
```
The option specified is not supported.

```
Failure: ICS<n> at device address <nn> failed operation.
```
The COMM_CONTROLLER command was not successful.

```
Failure: ICS<n> at device address <nn> failed prom self-verify operation.
```
Initialization of an ICS controller at device address *nn* has failed.

```
Failure: ICS<n> at device address <nn> failed downline load operation.
```
Down line loading of an ICS controller at device address *nn* has failed.

```
Failure: ICS<n> at device address <nn> failed load start packet operation.
```
An ICS controller at device address *nn* was unable to restart after loading.

```
Failure: ICS<n> at device address <nn> failed I/O bus initialization.
```
Communication cannot be activated to an ICS controller at device address *nn*.

```
Failure: ICS<n> at device address <nn> failed upline dump operation.
```
Unable to copy the contents of microprocessor prom from an ICS controller at device address *nn*.

3

USEFUL CONFIGURATION DIRECTIVES

The PRIMOS operating system is configured at every cold start based on parameters stored in the CONFIG file. Configuration directives define these parameters, fine tune the system, and allocate resources where they are needed most. This chapter describes the configuration directives that define and control communication.

Configuration directives perform five basic functions:

● Control the operating system

● Set internal timers

● Establish internal tables

● Inventory hardware

● Allocate memory

It is your responsibility as System Administrator to design a viable configuration that allocates resources, such as memory, to each user and establishes communication between PRIMOS and all devices. If you have any questions concerning the PRIMOS.COMI and CONFIG files, consult the *System Administrator's Guide, Volume I: System Configuration.*

Fifty-two configuration directives are currently supported. A complete list of all configuration directives is provided for your convenience in Table 3-1. Each directive is accompanied by a brief description and any default values. The directives marked with the checkmark symbol √ are discussed further in this chapter. For more information on these and other directives that are needed to operate your system, see the *System Administrator's Guide, Volume I: System Configuration.*

TABLE 3-1.   PRIMOS Configuration Directives

Directive	Description	Default Value
ABBREV	Permits user defined abbreviations	YES
√ AMLBUF	Sets the capacity of I/O buffers for asynchronous lines	IRB 200₈ (128) ORB 300₈ (192) DMQ 40₈ (32)
√ AMLCLK	Sets the speed of the software programmable clock for the AMLC and ICS hardware	22600₈ (9600)
√ AMLIBL	Sets the size of the DMC AMLC input tumble tables	60₈ (48)
√ AMLTIM	Sets time intervals for data set signal management	Ticks .2 sec. Disc 3 min. Grace 0
√ ASRATE	Sets the baud rate of the supervisor terminal	2010₈ (300)baud
√ ASRBUF	Sets the capacity of I/O buffers for the supervisor terminal	IRB 200₈ (128) ORB 300₈ (192)
√ ASYNC JUMPER	Defines three alternate speeds for asynchronous lines	113₈ (75) 226₈ (150) 3410₈ (1800)
COMDEV	Specifies the physical device number of the system command device	Site-specific
COMDVM	Activates disk mirroring of the command device	NO
√ DISLOG	Automatically logs out disconnected users	NO
√ DTRDRP	When included in the CONFIG file, drops the Data Terminal Ready signal when users log out	DTRDRP Enabled
ERASE	Sets the system default erase character	ASCII 242 (")
FILUNT	Guarantees the number of file units available for each user	77772₈ (32762)
GO	Indicates the last non-comment line of the CONFIG file	Required
√ ICS CARDS	Inventory, in octal, the quantity and location of asynchronous ICS line adapter cards	Site-specific
√ ICS INPQSZ	Sets the size of all the ICS input queues	77₈ halfwords
√ ICS INTRPT	Sets the I/O interrupt rate for ICS controllers	12₈ (10)

TABLE 3-1.   PRIMOS Configuration Directives (Continued)

Directive	Description	Default Value
KILL	Sets the system default kill character	*ASCII 277 (?)*
√ **LHC**	Assigns physical addresses to LHC controllers	*Site-specific*
LOGBAD	Displays unsuccessful login attempts at the supervisor terminal	*NO*
LOGLOG	Allows the LOGIN command to be issued in succession to perform an implicit LOGOUT	*YES*
LOGMSG	Displays login and logout messages at the supervisor terminal	*YES*
LOTLIM	Specifies login time limit in minutes	*3*
LOUTQM	Sets the number of minutes before an inactive user is logged out	*1750_8 (16 hours, 40 minutes)*
MAXPAG	Allocates a number of pages for physical memory	*All available memory*
MEMHLT	Halts the system when a two-bit memory parity error occurs	*YES*
MIRROR	When included in the CONFIG file, activates disk mirroring of the named file system partitions	*MIRROR enabled*
√ **NAMLC**	Reserves buffers for assignable asynchronous lines	*0*
NLBUF	Specifies the number of LOCATE buffers	*100_8 (64)*
√ **NPUSR**	Sets the number of phantom users	*4*
√ **NRUSR**	Allocates processes and workspace for PRIMENET remote login users	*0*
NSEG	Specifies the total virtual address space for a system	*1776_8 (1022)*
√ **NSLUSR**	Sets the number of slave processes	*0*
√ **NTSABF**	Sets the size of I/O buffers for NTS attached assigned lines	*IRB 200_8 (128) ORB 300_8 (192)*

TABLE 3-1.   PRIMOS Configuration Directives (Continued)

Directive	Description	Default Value
√ NTSASL	Reserves sets of I/O buffers for NTS attached assigned lines	0
√ NTSBUF	Sets the size of I/O buffers for NTS terminal users	*IRB 200$_8$ (128)* *ORB 300$_8$ (192)*
√ NTSUSR	Specifies the number of NTS terminal users	0
√ NTUSR	Specifies the total number of terminal users, including the supervisor terminal	*Site-specific*
NVMFS	Sets the number of Virtual Memory File Access (VMFA) dynamic segments available in virtual address space for the system	*144$_8$ (100)*
PAGING	Allocates as many as eight named paging partitions	*Site-specific*
PAGINM	Activates disk mirroring of the named paging device	*Site-specific*
√ REMBUF	Sets the sizes of I/O buffers for remote users	*IRB 202$_8$* *ORB 101$_8$*
√ SYNC CNTRLR	Maps a physical controller to a logical device address and selects a downline protocol	*ASYNC*
√ SYNC DSC	Defines Data Set Control actions performed by DPTX/BSCMAN for a logical line provided by an SMLC, HSSMLC, or MDLC controller	*line 1 2 0*
√ SYNC ON	Configures all synchronous communication drivers	*Site-specific*
√ SYNC SYNCnn	Maps a logical line number to a physical line number on a specified controller	*Site-specific*
SYSNAM	Specifies the system name at cold start	*Site-specific*
TPDUMP	When included in the CONFIG file, allows a crash dump before performing a forced shutdown	*TPDUMP enabled*
TYPOUT	Displays configuration directives at the supervisor terminal	*NO*
UPS	Controls restart after a power failure	*177777$_8$ (NO)*
WIRMEM	When included in the CONFIG file, prints the size of wired memory at the supervisor terminal during cold start	*Included*

Configuration Directive Dictionary

This section describes the configuration directives that define, enable, and regulate communication between PRIMOS and devices. For more information on these and other directives, see the *System Administrator's Guide, Volume I: System Configuration.*

Note

All numerical arguments to configuration directives must be octal numbers. Calculate the appropriate values as you normally would in decimal and Use the octal conversion function of the TYPE command, as shown below, to display the octal equivalent of the decimal values.

    OK, TYPE [TO_OCTAL 64]

    100

▶ **AMLBUF**

Sets the size of I/O buffers for local login and assignable asynchronous lines. AMLBUF directives are necessary only for lines connected to devices that do not run efficiently with the default buffer sizes. This directive has two distinct formats as shown below. Each login line requires a separate AMLBUF directive. Assignable lines may require two directives. If any of the size arguments to the AMLBUF directive are set to zero or omitted, the default size or, when there is an existing AMLBUF directive, the last value specified is used.

For login lines, only one AMLBUF directive is necessary:

    AMLBUF line-number   IRB-size   ORB-size   DMQ-size

Assignable lines may require two AMLBUF directives. The first AMLBUF directive sets only the DMQ size and the second sets the ring buffer sizes.

    AMLBUF line-number   0   0   DMQ-size
    AMLBUF index   IRB-size   ORB-size   0

It is not possible (except at cold start) to predetermine which ring buffers a particular assigned line receives. Therefore, it is recommended that you set the ring buffers for *all* assignable lines to the largest size that any individual device requires.

The AMLBUF directive cannot modify I/O buffer sizes for the supervisor terminal, NTS assignable lines, NTS terminal users, or PRIMENET remote login users. Use the ASRBUF, NTSABF, NTSBUF and REMBUF directives, respectively, for these users.

The arguments to the AMLBUF directive are as follows:

Argument	Description
line-number	The physical line number required for setting the DMQ buffer.
index	The value used by PRIMOS to reference ring buffers for assignable lines. *index* ranges from (NTUSR + NRUSR -1) to (NTUSR + NRUSR -1 + NAMLC).
IRB-size	The size in halfwords (two characters per halfword) of the input ring buffer. The minimum value is 1 and the maximum is 7777_8 (4095). The default value is 200_8 (128).
ORB-size	The size in halfwords (two characters per halfword) of the output ring buffer. The minimum value is 62_8 (50) and the maximum is 7777_8 (4095). The default value is 300_8 (192).
DMQ-size	The size in halfwords (one character per halfword) of the direct memory queue buffers. The value must be a power of 2 ranging from 20_8 (16) through 2000_8 (1024), inclusive. If you specify a value less than 20_8 (16), the system sets the size to 20_8 (16). The default value is 40_8 (32).

When invalid or excessive values are supplied, these values are rejected and one of the following error messages is displayed at the supervisor terminal.

BAD DMQ AMLC CONFIGURATION.
    This message is displayed when individual buffers exceed 7777_8 (4095) halfwords.

BAD LINE <line number> IN AMLBUF COMMAND. (AMLBUF)
    This message is displayed when the octal *line number* supplied in the AMLBUF command is not a valid line number.

NO ROOM. AMLBUF (TFLADJ)
    This message is displayed when the total size of input buffers plus output buffers exceeds 768,000 (decimal) halfwords.

UNABLE TO INTIALIZE DMQ AMLC. (AINIT)
    This message is displayed when the sum of all DMQ buffers on the system exceeds 64,000 halfwords.

For an explanation of how I/O buffers are used in PRIMOS and sample AMLBUF directives, see Chapter 6, Allocating I/O Buffers, pages 6-2 and 6-7.

▶ **AMLCLK baudrate**

Sets the speed of the software programmable clock in the AMLC hardware.

Argument   *Description*

baudrate   Sets the clock speed to the number of bits per second. The minimum value for *baudrate* is 35_8 (29), the maximum is 45400_8 (19200), and the default value is 22600_8 (9600).

The default speed is recommended if you are using Auto Speed Detect (ASD) on any line. If you have ICS or LHC controllers, select a *baudrate* from the table of valid speeds for the ASYNC JUMPER directive.

▶ **AMLIBL DMC-size**

Sets the size of the DMC input tumble tables. Each AMLC controller multiplexes the input from all the lines it supports into a single pair of DMC tumble tables. If the AMLIBL directive is omitted from the configuration file, the default buffer size is 60_8 (48).

Argument   *Description*

DMC-size   The size in halfwords (one character per halfword) of the Direct Memory Circuit (DMC) tumble tables. When the AMLIBL directive is set to 0, the total amount of memory available for tumble tables is divided equally between all AMLC controllers. The minimum explicitly set value is 20_8 (16). The maximum value is variable; it depends on the number of AMLC controllers configured, and the amount of space available for buffers.

If the value supplied for *DMC-size* is out of range, one of the following error messages is displayed at the supervisor terminal during cold start.

BAD AMLIBL PARAMETER. (CINIT)
    This error message is displayed when the *DMC-size* specified is less than 20_8 (16) halfwords.

INPUT BUFFERS TOO LARGE. (AMINIT)
    This message is displayed when the total *DMC-size* specified for all the DMC tumble tables on the system is greater than the amount of available memory.

▶ **AMLTIM ticks disctime gracetime**

Sets time intervals for the three variable event timers used in data set signal management.

The arguments have the following values and meanings:

Argument	*Description*
ticks	The interval (in tenths of a second) between carrier check operations. At the end of each period, PRIMOS checks each line for carrier signal loss. If a loss has occurred and the DISLOG directive is set, the process is logged out. The value for *ticks* must be greater than 0. The default is 2 (0.2 seconds).
disctime	The time period (in tenths of a second) after a disconnect when the DTR signal is again raised on all lines. A modem that was forced to be inactive because the DTR signal was dropped can establish an active carrier again. (The most common reason for a carrier to be inactive is that a dialup line has been hung up.) Specifying a value of 0 disables this feature. Otherwise, the value must not be less than the value of *ticks* and is truncated to the nearest multiple of that value. The default is 3410_8 (1800 decimal, which is 3 minutes). For NTS users the disctime is ignored.
gracetime	The minimum grace period (in tenths of a second) for a terminal line that has an active carrier but is not connected to a logged-in process. *gracetime*, in effect, defines the minimum time for a caller or NTS user to establish a logged-in process. (The actual grace period varies from *gracetime* to twice *gracetime*.) The default value of 0 disables the grace period. The specified value (if not 0) must be greater than *ticks* and is truncated to the nearest multiple of *ticks*. The value of *gracetime* should be large enough to enable PRIMOS to generate a forced logout of a previous user and enable another user to complete a login attempt.

Note

The AMLTIM directive affects the operation of Auto Speed Detect (ASD). No standard settings for the AMLTIM parameters can be recommended if your installation has ASD, but the following values for *ticks*, *disctime*, and *gracetime* have been shown to give satisfactory results. Set the value for *ticks* to at least 24_8. Set the value for *disctime* to at least twice that of *ticks*, preferably larger than 400_8. Set a value of 1000_8 for *gracetime*. ASD uses up a portion of *gracetime* before a user logs in.

▶ **ASRATE** rate

Sets the baud rate of the supervisor terminal. The *only* valid bitstrings for *rate* are shown below with the corresponding baud rate.

Bitstring	Baud Rate (Decimal)
0110	110
1010	300 (default)
2010	1200
3410	9600

ASRATE ensures that any subsequent configuration error messages are transmitted to the supervisor terminal at the correct speed. It should be the first directive in the configuration file.

▶ **ASRBUF IRB-size ORB-size**

Sets the sizes of the supervisor terminal I/O buffers.

The arguments have the following values and meanings:

Argument	*Description*
IRB-size	The size in halfwords (two characters per halfword) of the supervisor terminal input ring buffer. The default is 200_8 (128). If 0 is specified, the buffer size remains at its previously set value (which is usually the default size).
ORB-size	The size in halfwords (two characters per halfword) of the supervisor terminal output ring buffer. The default is 300_8 (192). The minimum value (other than 0) is 100_8 (64). If 0 is specified, the buffer size remains at its previously set value (which is usually the default size).

Noticeable delays occur when frequent messages are sent to a supervisor terminal that has standard *ORB-size* and is operating at a relatively slow baud rate. To correct this situation, enlarge the *ORB-size*.

▶ **ASYNC JUMPER speeda speedb speedc**

Defines three alternate line speeds for asynchronous lines. If the ASYNC JUMPER directive is not specified, the default values for *speeda*, *speedb*, and *speedc* are 113_8 (75), 226_8 (150), and 3410_8 (1800) bps, respectively. You can choose other values from the following list. The speeds you can use on lines configured for Auto Speed Detect (ASD) are marked with an asterisk. ICS lines and NTS assignable lines support all of the speeds listed.

AMLC lines, however, support ASYNC JUMPER speeds only when they are configured for ASD. The default speeds are wired at the factory with small jumper cables on each AMLC controller board. When the default speeds are changed, these jumper cables must be rewired. The speeds specified in the ASYNC JUMPER directive must agree with the hardwired speeds on the controller board.

The ASYNC JUMPER directive makes the same three alternate speeds available for all controllers on your system. Which three speeds you choose is up to you, but when these speeds are used for AMLC lines, as well as NTS or ICS lines, they must match the hardware jumper speeds on the AMLC boards.

Speed (bps)	*Speed (bps)*
62_8 (50)	3410_8 (1800)
113_8 (75)	4540_8 (2400)*
156_8 (110)*	7020_8 (3600)
226_8 (150)	11300_8 (4800)*
310_8 (200)	16040_8 (7200)
454_8 (300)*	22600_8 (9600)*
1130_8 (600)*	45400_8 (19200)*
2260_8 (1200)*	program AMLCLK

In the following example, an ICS3 controller supports dialup lines used by 300 and 1200 baud modems, as well as a 19200 baud terminal used to transmit asynchronous characters to a PAD to form standard X.25 data packets. Set the ASYNC JUMPER directive in the PRIMOS.COMI file to make these three speeds available systemwide. Then with the SET_ASYNC command, configure the individual lines to recognize and use the jumper speeds.

```
ASYNC JUMPER 454 2260 45400
     /*Set jumper speeds to 300, 1200, and 19200 bps
SET_ASYNC -LINE 155 -PRO TTY -SPEED_DETECT -SPEED speeda
     /*300 baud modem
SET_ASYNC -LINE 163 -PRO TTY -SPEED_DETECT -SPEED speedb
     /*1200 baud modem
SET_ASYNC -LINE 178 -PRO TTY -SPEED_DETECT -SPEED speedc
     /*19200 terminal
```

▶ **DISLOG** value

Enables or disables automatic logout when a line is disconnected. When the incoming Data Carrier Detect (DCD) signal drops or becomes logically low, a line is assumed to be disconnected.

The DISLOG directive is particulary useful for installations with port selectors or dialup modems. A security problem can exist if a user is disconnected without logging out of PRIMOS. The next user to dial that line is actively logged into the system without going through any validation procedures, such as a login password, or any additional site-specific procedures. To prevent this from happening, specify DISLOG YES or DISLOG line_number.

value can have the following meanings:

value	*Description*
NO	Does not log out the user of any line if the user's line is disconnected. (Default)
YES	Logs out any user whose line is disconnected. YES is recommended if you are using ASD systemwide.
line_number	Logs out only the user of the line specified when the line is disconnected, where *line-number* is the octal line number. Individual lines require a separate DISLOG directive. Per-line DISLOG is recommended for dialup lines using ASD connected to modems or port selectors.

The SET_ASYNC command can also provide per-line DISLOG functionality. See Chapter 4, Configuring Asynchronous Lines, for more information on the SET_ASYNC -DISLOG and -NO_DISLOG options.

The following restrictions apply to the DISLOG directive:

● One DISLOG directive that follows another cannot override that first directive. For example, DISLOG YES followed by DISLOG NO causes every line to be set for DISLOG.

● A global DISLOG directive takes precedence over a per-line directive. For example, when there are several per-line directives and a DISLOG YES directive, DISLOG is enabled on every line.

● NTS lines are automatically logged out on disconnect whether or not DISLOG is enabled.

▶ DTRDRP

PRIMOS raises a Data Terminal Ready (DTR) signal on dialup lines to permit the modem or port selector to automatically answer the next user. After the dialup connection is made, the DTR signal is maintained until a user logs out, or hangs up the phone. When the connection is broken, PRIMOS briefly drops the DTR signal to terminate the session and then raises it to make the line available for the next user.

If specified, the DTRDRP directive automatically forces the dropping of the DTR for any user when that user logs out, regardless of the period set by the *gracetime* value of the AMLTIM directive.

DTRDRP is useful only for installations using Auto Speed Detect (ASD) with port selectors or dialup modems. (Users who have logged out can also issue the PRIMOS DROPDTR command explicitly.)

▶ ICS CARDS device-address config-word

Inventories the quantity and location of the asynchronous Line Adapter Cards (LACs) for each ICS2 or ICS3 controller. PRIMOS uses the ICS CARDS directives to verify the actual configuration of LACs at both cold and warm starts.

ICS controllers maintain line numbers and configuration information for asynchronous lines from cold start to shutdown whether or not this directive is used. For example, line 121 will always be line 121 regardless of whether a LAC is removed. However, the ICS CARDS directive verifies that the asynchronous configuration has not changed. If the ICS CARDS directive is omitted for any ICS2 or ICS3 on your system, its configuration is not checked at cold start.

When you add or remove a LAC, you must change the ICS CARDS directive for that controller, specifically the *config-word*.

The arguments have the following values and meanings:

Argument	*Description*
device-address	The address of the ICS2 or ICS3 controller. Valid addresses are 10_8, 11_8, 36_8, and 37_8.
config-word	An octal map of an ICS Card Cage on which each bit corresponds to one of the sixteen slots. A bit with a value of 1 means that an asynchronous LAC is present in that slot. A bit with a value of 0 means that either a synchronous LAC is present in that slot or the slot is empty.

Use the following four-step process to calculate the *config-word*:

1. Fill in the ICS Card Cage Inventory in Figure 2-1. If a slot contains an asynchronous LAC, enter 1 in that position. If a synchronous LAC is present or the slot is empty, enter zero.

2. Use the table to convert this binary number to octal.

3. Write the octal equivalent for each block in the spaces provided. The result is the *config-word*.

4. Repeat steps 1 through 3 for each LAC Card Cage on your system.

For further details on ICS input queues, see the *ICS User's Guide*.

ICS CARDS Error Messages: The ICS CARDS directive must agree with the actual physical configuration at cold or warm starts. When the directive detects unexpected, faulty, or missing LACs, an error message is displayed. Each message contains the controller's device address *dd* and the specific slot number *ss*.

```
ICS device <dd> has returned the wrong number of status words.

Inconsistent ASYNC cold start configuration for ICS device <dd>:
an async line card has been found where not expected in slot <ss>.

Inconsistent ASYNC cold start configuration for ICS device <dd>:
faulty line card in slot <ss> where an async line card was expected.

Inconsistent ASYNC cold start configuration for ICS device <dd>:
sync line card in slot <ss> where an async line card was expected.

Inconsistent ASYNC cold start configuration for ICS device <dd>:
slot <ss> is empty where an async line card was expected.

Inconsistent ASYNC warm start configuration for ICS device <dd>:
an async line card has been inserted into slot <ss>.

Inconsistent ASYNC warm start configuration for ICS device <dd>:
the async line card in slot <ss> is now inoperable.

Inconsistent ASYNC warm start configuration for ICS device <dd>:
the async line card in slot <ss> has been removed or is now inoperable.

Inconsistent ASYNC warm start configuration for ICS device <dd>:
a sync line card has been inserted into slot <ss>.
```

ICS Card Cage
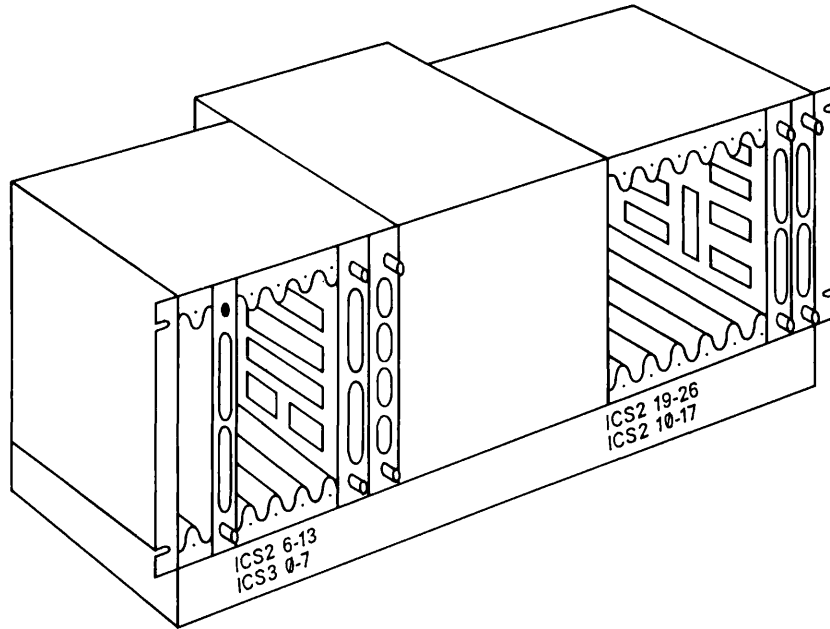
ICS Card Cage Inventory

ICS2 Slot Numbers	6	7	8	9	10	11	12	13	19	20	21	22	23	24	25	26
ICS3 Slot Numbers	0	1	2	3	4	5	6	7	10	11	12	13	14	15	16	17

0 = SYNC LAC or Empty Slot

1 = ASYNC LAC
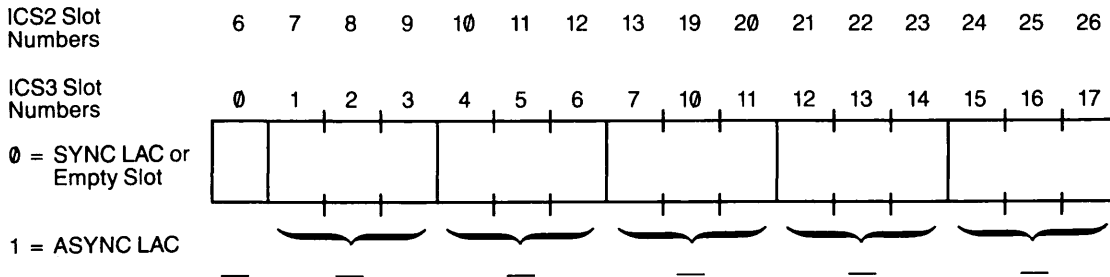
Conversion Table

BINARY	000	001	010	011	100	101	110	111
OCTAL	0	1	2	3	4	5	6	7

FIGURE 3-1. ICS CARDS Directive Template

▶ **ICS INPQSZ DMQ-size**

Sets the size of all the ICS input queues, one pair for every eight lines. In general, the default size is adequate unless either block mode terminals are used, or memory is limited. When an invalid DMQ-size is supplied, cold start fails and an error message is displayed at the supervisor terminal. The argument DMQ-size has the following value and meaning:

Argument   *Description*

DMQ-size   The size in halfwords, (one character per halfword) of the DMQ buffers. *DMQ-size* must be less than 1777_8 and equal to one less than a power of two. The default is 77_8 halfwords (63). Examples of possible queue sizes are 17_8, 37_8, 77_8, 177_8, 377_8, and 777_8.

▶ **ICS INTRPT rate**

Sets the asynchronous interrupt rate for ICS controllers. The argument has the following value and meaning.

Argument   *Description*

rate   Specifies the number of interrupts per second. The minimum value is 12_8 (10) (default). The recommended value is 36_8 (30). The maximum value is 144_8 (100). The interrupt rate must be in multiples of 10 milliseconds.

Calculating Interrupt Rates: In most cases it is not necessary to increase the ICS interrupt rate. If increased buffer sizes do not correct perceptible bursts of output for block mode devices and full-screen graphics packages, increase the interrupt rate to improve performance. For more information on I/O interrupt rates read the section, The ICS I/O Interrupt Rate, on page 6-5. To set a value between 12_8 and 144_8, use Table 3-2.

TABLE 3-2.   ICS Interrupt Rates

Interrupt Rate	*Decimal Equivalent*	*Time Between Interrupts (ms)*
12_8	10	100
13_8	11	90
14_8	12	80
$15_8, 16_8$	13,14	70
$17_8, 20_8$	15,16	60
21_8-24_8	17-20	50
25_8-31_8	21-25	40
32_8-41_8	26-33	30
42_8-62_8	34-50	20
63_8-144_8	100	10

When the value supplied for the ICS INTRPT rate is less than the minimum or greater than the maximum, the ICS INTRPT directive substitutes the minimum value, 12_8, or the maximum value, 144_8, as appropriate, and cold start continues. In both cases the following message is displayed.

```
BAD ICS DIRECTIVE: INTRPT
```

▶ **LHC number address**

Sets the address assignment for a LAN300 Host Controller (LHC) to agree with its logical controller number. The most recent LHC directive overrides any existing address assignment.

The arguments have the following values and meanings:

Argument	*Description*
number	Indicates the logical number assigned to the LHC in the NTS configuration file. *number* ranges from 0 through 7. This number is also used in the PRIMENET and NTS configuration files.
address	Specifies the LHC's physical device address in octal. Valid addresses are 10_8, 11_8, 15_8, 16_8, 17_8, 32_8, 35_8, 36_8, 37_8, 50_8, 51_8, 52_8, 53_8, 54_8, and 56_8.

Use the LIST_COMM_CONTROLLERS command to display the current logical controller number and the octal device address. You can obtain the two-digit octal device address with the STATUS COMM command, as well. The LHC directive detects several error conditions, typically missing or out-of-range arguments. When an error is recognized, the directive is aborted and, unless indicated below, no action is taken.

```
Bad LHC directive, LHC number must be between 0-7.
```
This message is displayed when the logical controller number is out of range.

```
Bad LHC directive, missing arguments.
```
This message is displayed when either one or both numerical arguments are omitted.

```
LHC directive ignored, device at <nn> is not an LHC.
```
This message is displayed when an attempt is made to associate an LHC with a device address *nn* that is currently being used by another type of controller.

```
LHC directive ignored, no LHC at address <nn>.
```
This message is displayed when an attempt is made to associate an LHC with a physical device address *nn* that is currently not in the system.

```
LHC <n> overrides LHC <m> with address <nn>.
```
This warning message is displayed to alert the System Administrator when more than one LHC is mapped to a single physical device address. Remember, the most recent directive takes precedence over any existing directives for LHC *m*, and the LHC *n* is assigned to address *nn*.

▶ **NAMLC number**

Reserves one pair of IRB and ORB buffers for each local assignable asynchronous line. NAMLC + NTUSR + NTSUSR + NTSASL must be less than or equal to 377_8 (256). The argument *number* has the following value and meaning:

Argument   *Description*

number   The maximum number of local assignable lines that can be used simultaneously. The default is 0.

▶ **NPUSR number**

Sets the number of phantom users. At Rev. 21.0 at least 10 or more phantoms are recommended. The argument *number* has the following value and meaning.

Argument   *Description*

number   The total number of phantom users available. The default and minimum is 4. The maximum is 377_8 (255) minus the number of terminal, slave, remote, and NTS users (NTUSR, NSLUSR, NRUSR, and NTSUSR).

You must configure phantoms for the LOGIN_SERVER and the TIMER_PROCESS. If you have PRIMENET, you must configure a phantom user for NETMAN. If your system is to be a gateway node, you must also configure a phantom for the RT_SERVER. Phantoms are also required for DSM, NTS, the Spooler subsystem, the Batch subsystem, and the File Transfer Service (FTS). Whenever you install a new product, refer to the product's installation guide to determine if it requires a phantom.

▶ **NRUSR number**

Specifies the number of processes to be reserved for remote logins across the PRIMENET network. Dialup users who connect through modems to your system are considered local users and are not included in NRUSR. Similarily, NTS users are not included in this total. The argument *number* has the following value and meaning:

Argument   *Description*

number   The total number of concurrent remote users. The default and minimum is 0. The maximum is 77_8 (63). NTUSR + NPUSR + NRUSR + NSLUSR + NTSUSR cannot exceed 377_8 (255).

Note

The total number of virtual circuits available for remote and slave users combined is increased to 249 at Rev. 21.0. You can configure up to 127 remote log through users as well. However, only 249 can be active at one time. An attempt to exceed that limit produces an error message stating that the resource is temporarily unavailable.

▶ **NSLUSR number**

Specifies the number of slave processes or other NPX users available to access files on a local system for remote users. Each remote user requires a local slave process for the duration of the session. NTS users must configure at least one slave to receive messages from operators on other nodes. The argument *number* has the following value and meaning:

Argument	*Description*
number	The total number of simultaneous remote file accesses that the local system supports. The default and minimum value is 0. The maximum value is 77_8 (63). NTUSR + NPUSR + NRUSR + NSLUSR + NTSUSR must be less than or equal to 377_8 (255).

If all slave users are assigned when a remote user makes an attach request, the following error message is displayed.

```
No NPX slaves available (ATTACH)
```

▶ **NTSABF line IRB-size ORB-size XOFF-lag XON-lag**

Sets the size of I/O buffers and flow control thresholds for NTS assignable lines connected to the local system. If an argument is omitted, the default value is used, unless a previous NTSABF directive still exists; then the last value specified is used.

The arguments have the following meanings and default values.

Argument	*Description*
line	The only legal value for Rev. 21.0 is 0. This reserves a standard set of I/O buffers for all NTS assignable lines.
IRB-size	The size in halfwords (two characters per halfword) of the input ring buffer. The minimum value is 1, the maximum is 7777_8 (4095), and the default value is 200_8 (128).
ORB-size	The size in halfwords (two characters per halfword) of the output ring buffer. The minimum value is 62_8 (50), the maximum is 7777_8 (4095), and the default value is 300_8 (192).

Argument	Description
XOFF-lag	The threshold point in the IRB when an XOFF is sent to the line to stop the incoming stream of characters. XOFF-lag is measured in halfwords and must be less than the IRB-size. The default value is 60% of the IRB-size.
XON-lag	The threshold point in the IRB when an XON is sent to the line to restart the incoming stream of characters. XON-lag is measured in halfwords and must be less than the IRB-size. The default value is 20% of the IRB-size.

When *line* is not 0, the specific line number is ignored and the following message is displayed:

```
Warning: NTS line number must be zero at this release. (NTSABF)
```

The total size of all IRBs and all ORBs on the system can not exceed 768,000 halfwords. Exceeding this limit produces the following message:

```
NO ROOM. NTSABF (TFLADJ).
```

If the NTSABF directive is issued when NTSASL is 0, the following message is displayed:

```
Warning: NTSABF directive issued with no NTS assigned lines configured.
(CINIT)
```

▶ **NTSASL number**

Reserves one pair of IRB and ORB buffers for each NTS assignable line. The argument *number* has the following value and meaning.

Argument	Description
number	The maximum number of NTS assignable lines that can be used simultaneously on your system. The default is 0. NAMLC + NTSASL + NTUSR + NTSUSR must be less than or equal to 377_8 (255).

Note

When either NTSUSR or NTSASL is a nonzero integer, the NTS connection manager, NTS_SERVER, is started. Include an additional phantom for this process in the NPUSR directive. If a phantom is not available at cold start for the NTS_SERVER, NTS cannot start.

▶ **NTSBUF line IRB-size ORB-size XOFF-lag XON-lag**

Sets the size of I/O buffers and flow control thresholds for NTS terminal users connected to the local system. If an argument is omitted or set to zero, the default value, or, when there is an existing NTSBUF directive, the last value specified is used.

The arguments have the following meanings and default values:

Argument	*Description*
line	The only legal value for Rev. 21.0 is 0. This reserves sets of standard I/O buffers for all NTS terminal users.
IRB-size	The size in halfwords (two characters per halfword) of the input ring buffer. The minimum value is 1, the maximum is 7777_8 (4095), and the default value is 200_8 (128).
ORB-size	The size in halfwords (two characters per halfword) of the output ring buffer. The minimum value is 62_8 (50), the maximum is 7777_8 (4095), and the default value is 300_8 (192).
XOFF-lag	The threshold point in the IRB when an XOFF is sent to the line to stop the incoming stream of characters. XOFF-lag is measured in halfwords and must be less than the IRB-size. The default value is 60% of the IRB-size.
XON-lag	The threshold point in the IRB when an XON is sent to the line to restart the incoming stream of characters. XON-lag is measured in halfwords and must be less than the IRB-size. The default value is 20% of the IRB-size.

When *line* is not 0, the specific line number is ignored and the following message is displayed:

```
Warning: NTS line number must be zero at this release. (NTSBUF)
```

The total size of all IRBs and all ORBs on the system cannot exceed 768,000 halfwords. Exceeding this limit produces the following message:

```
NO ROOM. NTSBUF (TFLADJ).
```

If the NTSBUF directive is issued when NTSUSR is 0, the following message is displayed:

```
Warning: NTSBUF directive issued with no NTS terminal users configured.
(CINIT)
```

▶ **NTSUSR number**

Specifies the maximum number of simultaneous NTS terminal users. The argument *number* has the following value and meaning.

Argument   *Description*

number   The total number of NTS terminal users. The default is 0. NTUSR +
                    NPUSR + NRUSR + NTSUSR + NSLUSR must be less than or equal to
                    377_8 (255).

Note

When either NTSUSR or NTSASL is a nonzero integer, the NTS connection manager, NTS__SERVER, is started. Include an additional phantom for this process in the NPUSR directive. If a phantom is not available at cold start for the NTS__SERVER, NTS cannot start.

▶ **NTUSR number**

Specifies the number of local terminal users, including the supervisor terminal. NTS terminal users are not included in this total. NTUSR must be included in your CONFIG file. The argument *number* has the following value and meaning.

Argument   *Description*

number   The number of local terminal users for which the system is to be
                    configured. The minimum value of *number* is 2_8 and the maximum is
                    377_8 (255). NTUSR has no default value. NTUSR + NPUSR + NRUSR +
                    NSLUSR + NTSUSR must be less than or equal to 377_8 (255).

▶ **REMBUF IRB-size ORB-size**

Sets the sizes of the terminal input and output buffers for remote users. The arguments have the following values and meanings:

Argument   *Description*

IRB-size   The terminal input buffer size in halfwords (two characters per
                    halfword). The default and minimum value is 202_8 halfwords (260
                    bytes decimal). If 0 is specified, the buffer size remains at its previously
                    set value (which is usually the default size).

ORB-size   The terminal output buffer size in halfwords (two characters per
                    halfword). The default and minimum value is 101_8 halfwords (130
                    bytes decimal). If 0 is specified, the buffer size remains at its previously
                    set value (which is usually the default size).

Total size of all input buffers plus all output buffers cannot exceed 2734000_8 (768,000) halfwords. On systems using block-mode terminals, you can improve throughput by increasing the size of the input buffer to 404_8 halfwords (520 bytes decimal). You can also improve throughput on systems where users are logging in remotely across a ring network by increasing the size of the output buffer to 402_8 halfwords (516 bytes decimal).

For remote login over Public Data Networks (PDNs), you must set *IRB-size* and *ORB-size* to the octal value of the packet size that you configured with CONFIG_NET. For example, a packet size of 200_8 (128) bytes requires that both *IRB-size* and *ORB-size* be set to 200_8.

▶ **SMLC CNTRLR**
▶ **SMLC DSC**
▶ **SMLC ON**
▶ **SMLC SMLCnn**

At Rev. 20.0, SYNC became the preferred synonym for SMLC. For details on the SMLC directives, see the discussions of the SYNC directives, below.

▶ **SYNC**

The four SYNC directives are detailed below. Here is a summary of their purposes:

SYNC CNTRLR	Associates a device address with a logical controller number and a communications protocol.
SYNC DSC	Specifies data set control information.
SYNC ON	Enables the configuration of synchronous communication drivers.
SYNC SYNCnn	Associates a controller's physical line with a logical line number.

Note

At Rev. 21.0 the new COMM_CONTROLLER command duplicates the functionality of the existing SYNC configuration directives. You can use either one or a combination of both. See Chapter 2, Operating Intelligent Controllers, for a complete description of the COMM_CONTROLLER command.

▶ **SYNC CNTRLR controller-number device-address protocol**

Assigns a physical controller address to a logical controller number with a particular protocol. This directive must be given before any SYNC SYNCnn directive. It enables an ICS2 or ICS3 to handle synchronous communications.

The arguments have the following values and meanings:

Argument   *Description*

controller-number

The logical controller. Valid values are 0, 1, or 7. Any other number produces the error message:

BAD SYNC CONTROLLER MAPPING COMMAND.

The value 7 is for SDLC or ASYNC_SDLC only and allows multiple controllers (that only support SDLC) to be configured. You can have more than one controller-number 7.

device-address   The physical device address of the specified controller given in octal. Usually 10_8 or 11_8 is given for an ICS2 or ICS3, and 36_8 or 37_8 for an ICS1. The default values are address 50_8 for controller 0 and 100000_8 (disabled address) for controller 1. Note that device address 50_8 is usually an MDLC. If you use controller 1, its address must not conflict with the address of any other peripheral controller.

protocol   One or more tokens, specifying communications protocols. The tokens select files to be downline loaded into an ICS2 or ICS3 controller. Use this argument only with an ICS2 or ICS3. See Tables 3-3 and 3-4 for valid protocol combinations.

Entering a protocol token for a controller other than an ICS2 or ICS3 controller causes the following error message to be displayed:

Error: Controller <xx> does not support sync protocols
        (COMINI)

where *xx* is the device address of the controller.

If the protocol token is omitted, the following error message appears:

Error: protocol combination not supported on ICS2
        device address <dd> (BTPCC).

Note

If the SYNC CNRTLR directive is omitted entirely, the default protocol (ASYNC) is loaded.

If you map one logical controller to a previously mapped address, SYNC automatically disables the previously mapped controller (without warning), setting its address to 100000_8. A new mapping directive enables the disabled controller. For example, the following directives map controller 1 to address 10_8 and controller 0 to address 11_8:

SYNC CNTRLR 1 10 ASYNC_HDLC
SYNC CNTRLR 0 11 BSCRJE_BSCX25

The *protocol* argument selects appropriate downline load files to support the specified protocol combination.

You can also disable a logical controller by setting its address to blank or to 100000_8. For example, either of the following directives can disable controller 0:

```
SYNC CNTRLR 0 100000
SYNC CNTRLR 0
```

Note that the second example above disables the default configuration for address 50_8.

Valid Protocol Token Combinations: Valid protocol tokens are ASYNC, SDLC, HDLC, BSCX25, and BSCRJE. You need not enter a token combination in the order shown in Tables 3-3 and 3-4, but the parts must be separated by an underscore.

ICS2: Table 3-3 lists valid protocol token combinations, microcode image names, and downline load file numbers for the ICS2. The ICS2 cannot support all protocols concurrently because of limitations to memory size.

> **Note**
>
> Asynchronous Reverse Flow Control (RFC) is not enabled for an ICS2 controller when you select a combination of BSC and ASYNC protocols (as in BSCRJE_BSCX25_ASYNC, BSCRJE_ASYNC, or ASYNC_BSCX25). If you use the AMLC command to attempt to enable RFC in this situation, the ICS2 ignores the request and PRIMOS returns an error message.

ICS3: The ICS3 is available with 256KB, 512KB, and 1024KB of RAM. In the 512KB ICS3, any combination of protocol tokens is allowed. When all protocols are required, the downline file ICS3_08.DL is loaded. The 256KB ICS3 can handle all protocol combinations except the two largest combinations, ASYNC_SDLC_HDLC_BSCX25_BSCRJE and SDLC_BSCX25_BSCRJE.

Any combination not containing SDLC causes downline load file ICS3_09.DL to be loaded. This file contains all protocols except SDLC. For any combination containing SDLC, the loaded file contains only SDLC and the other protocol (or protocols), resulting in more economical running than when loading the all protocols file, ICS3_08.DL.

Table 3-4 lists the protocol tokens, the microcode image names and the downline load file numbers used for the ICS3.

To conserve space, you can delete unused downline load files and maps from the DOWN_LINE_LOAD* directory.

TABLE 3-3.   ICS2 Valid Protocol Token Combinations

Protocol Token	Microcode Image[1]	DLL-file Number
ASYNC	HSAXX	ICS2_01.DL
SDLC	HSAXX	ICS2_02.DL
HDLC	HSAXX	ICS2_03.DL
BSCRJE_BSCX25	HSBXX	ICS2_04.DL
BSCX25	HSBXX	ICS2_04.DL
ASYNC_SDLC	HSAXX	ICS2_05.DL
ASYNC_HDLC	HSAXX	ICS2_06.DL
HDLC_SDLC	HSAXX	ICS2_07.DL
BSCRJE_BSCX25_ASYNC[2]	ABXX	ICS2_08.DL
ASYNC_BSCX25[2]	ABXX	ICS2_08.DL
BSCRJE_BSCX25_SDLC	HSBXX	ICS2_09.DL
BSCRJE_SDLC	HSBXX	ICS2_09.DL
BSCX25_SDLC	HSBXX	ICS2_09.DL
BSCRJE_BSCX25_HDLC	HSBXX	ICS2_10.DL
BSCRJE_HDLC	HSBXX	ICS2_10.DL
ASYNC_HDLC_SDLC	HSAXX	ICS2_11.DL
BSCRJE	HSBXX	ICS2_12.DL
BSCRJE_ASYNC[2]	ABXX	ICS2_13.DL
BSCX25_HDLC	HSAXX	ICS2_14.DL
BSCRJE_BSCX25_HDLC_SDLC	HSAXX	ICS2_15.DL
BSCX25_HDLC_SDLC	HSAXX	ICS2_15.DL
BSCRJE_HDLC_SDLC	HSAXX	ICS2_15.DL

---

[1]XX is the version number.

[2]No asynchronous Reverse Flow Control for this combination.

TABLE 3-4.   ICS3 Valid Protocol Token Combinations

Protocol Token	Microcode Image[3]	DLL-File Number
SDLC	ABHSXX	ICS3_01.DL
ASYNC_SDLC	ABHSXX	ICS3_02.DL
SDLC_HDLC	ABHSXX	ICS3_03.DL
SDLC_BSCX25	ABHSXX	ICS3_03.DL
SDLC_HDLC_BSCX25	ABHSXX	ICS3_03.DL
ASYNC_SDLC_HDLC	ABHSXX	ICS3_04.DL
ASYNC_SDLC_BSCX25	ABHSXX	ICS3_04.DL
ASYNC_SDLC_BSCX25_HDLC	ABHSXX	ICS3_04.DL
SDLC_BSCRJE	ABHSXX	ICS3_05.DL
ASYNC_SDLC_BSCRJE	ABHSXX	ICS3_06.DL
SDLC_HDLC_BSCRJE	ABHSXX	ICS3_07.DL
SDLC_BSCX25_BSCRJE	ABHSXX	ICS3_07.DL
SDLC_HDLC_BSCX25_BSCRJE	ABHSXX	ICS3_07.DL
ASYNC_SDLC_HDLC_BSCRJE_BSCX25	ABHSXX	ICS3_08.DL
ASYNC_SDLC_BSCRJE_BSCX25	ABHSXX	ICS3_08.DL
ASYNC_SDLC_HDLC_BSCRJE	ABHSXX	ICS3_08.DL
ASYNC_HDLC_BSCX25_BSCRJE	ABHSXX	ICS3_09.DL

▶ **SYNC DSC line strap proc recv**

Specifies data set control (DSC) information used by DPTX/BSCMAN for a logical line provided by an SYNC, HSSMLC, or MDLC controller. Use SYNC DSC for DPTX/BSCMAN only. If specified, SYNC DSC must appear after any SYNC SYNCnn directives. The arguments have the following values and meanings:

Argument   Description

line   The logical line number (00-07) represented by *nn* in the SYNC SYNCnn directive.

strap   A bit pattern that indicates specific data set signals to be strapped on by the software. Speed select (the 10_8 bit) is for European equipment. The bits for forming the pattern are as follows:

       01_8 Data Terminal Ready (DTR). The default is 1.

       02_8 Request to Send (RTS).

       10_8 Select fast data set. 00 selects slow data set. (Europe only.)

---

[3]XX is the version number.

Argument   Description

proc   The data set control procedure to be used for transmitting data. The procedures are as follows:

1   No data set orders. Usually used with DTR and RTS strapped on, with modems used for four-wire full-duplex service.

2   Use data set orders as follows: issue RTS, wait for CLEAR to send (CTS), send, drop RTS. Usually used with most half-duplex modems. The default is 2.

3   Use data set orders as follows: wait for .NOT. Carrier Detect (CD), issue RTS, wait for CTS, send, drop RTS. Rarely used, but may be necessary with 201-series modems only if lines are very noisy. Try 2 first.

recv   Indicates whether the receiver is to be turned on before or after transmitting. The settings are as follows:

0   Turn on receiver before transmitting. This setting provides a faster response and should be used if possible. The default is 0.

1   Turn on receiver after transmitting. This setting must be used with two-wire 201-series modems. This setting may be tried on other two-wire systems only if problems appear that cannot be solved by other means.

The default setup, if no DSC is specified, is the equivalent of including the following line in the configuration file:

```
SYNC DSC line 1 2 0
```

▶ **SYNC ON**

Configures synchronous communication drivers for all synchronous communication controllers, including the ICS2 and ICS3.

SYNC ON must be specified when synchronous lines are used for products other than PRIMENET. The default configuration for SYNC ON is given below.

Logical Line Number	*Logical Controller*	*Controller Address*	*Controller Physical Line Number*
00	0	50_8	0
01	0	50_8	1
02	0	50_8	2
03	0	50_8	3
04	1	100000_8	0
05	1	100000_8	1
06	1	100000_8	2
07	1	100000_8	3

Logical lines 00 to 03 are mapped to logical controller 0. The physical device address is 50_8, with physical line numbers 0 to 3. (Note that the MDLC device address is usually 50_8 or 51_8.)

Logical lines 04 to 07 are mapped to physical lines 0 to 3 on logical controller 1. Controller 1's physical address is 100000_8, indicating that the controller is disabled. To enable controller 1, set its address to a valid device address (50_8 or 51_8) with the SYNC CNTRLR directive.

You can change the default configuration with the SYNC CNTRLR and SYNC SYNCnn directives, which are described in this chapter. SYNC CNTRLR changes the mapping of the logical controller to the physical address. SYNC SYNCnn changes the mapping of a single logical line number.

You may use either directive separately. If you need both directives, you *must* give the SYNC CNTRLR directive(s) first. In other words, you must assign the controller its correct physical address before you assign any new SYNC lines to it.

Always specify synchronous line configuration explicitly. Do not rely on the default configuration for lines to be used or for lines to be left unused.

Be sure to override the default configuration when synchronous lines are configured on two controllers, one of which is not device address 50_8.

► **SYNC SYNCnn controller-number line-number**

Maps a logical line number to a physical line number on a specified logical controller.

SYNC SYNCnn is used for X.25/RJE, but not for SDLC. The arguments have the following values and meanings:

USEFUL CONFIGURATION DIRECTIVES

Argument	Description
nn	The logical line number. The values are 00 to 07.

controller-number

The logical controller set by a SYNC CNTRLR directive. The values are 0_8, 1_8, or 100000_8. Use 0_8 or 1_8 to identify a controller. Use 100000_8 when the specified line is not to be configured or allocated memory. (7 is not accepted.) The default is 100000_8.

line-number   The physical line number of the specified controller onto which the logical line number is mapped. (For ICS2 or ICS3 sync LAC physical line numbering, see the *ICS User's Guide*.) If the controller is an ICS1, *line-number* must be 0. The default values map SYNC00 to SYNC03 to physical lines 0 to 3 on the first controller, and SYNC04 to SYNC07 to physical lines 0 to 3 on the second controller. This value must be specified unless *controller-number* is 100000_8 or is unspecified.

For example, the following directive assigns logical line 4 to physical line 3 on controller 1:

```
SYNC SYNC04 1 3
```

Setting the controller number either to a blank or to 100000_8 disables a logical line number. For example, either of the following directives disables logical line 07:

```
SYNC SYNC07
SYNC SYNC07 100000
```

Giving any value for *controller-number* other than 0, 1, blank, or 100000, produces the following error message:

```
BAD SYNC LINE MAPPING COMMAND
```

3-29

CONFIGURING ASYNCHRONOUS LINES

Asynchronous communication is a method of transmitting data in which each character is sent separately. The time intervals between transmissions can vary. Therefore, each character is identified by a start bit and followed by one or more stop bits.

Asynchronous lines connect terminals, modems, and peripheral devices, such as printers and plotters, to your system. Figure 4-1 shows a typical system and the types of devices that communicate via asynchronous lines.

The first part of this chapter discusses how to use the SET_ASYNC command and its companion, the CONVERT_AMLC_COMMANDS utility, to configure your asynchronous communication lines. The second part provides background information on how to select protocols, determine line numbers, and use Auto Speed Detect.

For information on configuring synchronous lines for devices that use half-duplex or packet data exchanges, see the explanation of the SYNC directives in Chapter 3, Useful Configuration Directives.

Caution

At Rev. 20.2, the SET_ASYNC command replaced the AMLC command. SET_ASYNC provides a more straightforward way of configuring asynchronous lines. Although the AMLC command is still supported, its use is no longer recommended. For information on the AMLC command, see Appendix A, Obsolete and Rarely Used Commands.

If your PRIMOS.COMI file contains AMLC commands, use the CONVERT_AMLC_COMMANDS utility to translate any obsolete commands to their equivalent SET_ASYNC commands. Before upgrading your PRIMOS.COMI file read the two following sections, the SET_ASYNC Command on page 4-3, and the CONVERT_AMLC_COMMANDS Utility on page 4-16.
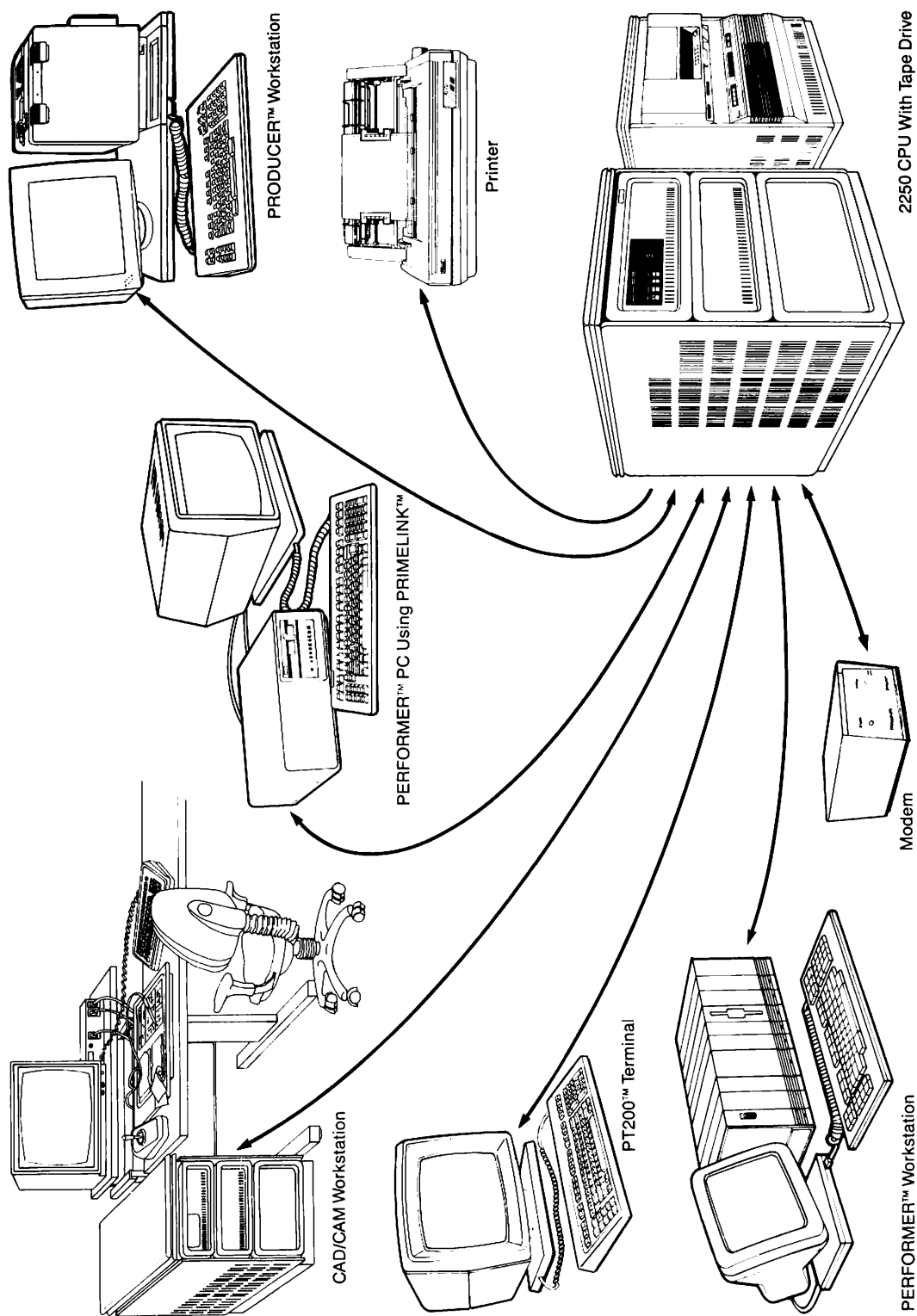
FIGURE 4-1.   Asynchronous Lines Connect Terminals and Peripherals to Your System

The SET_ASYNC Command

The SET_ASYNC command defines terminal line characteristics for an individual asynchronous line or a range of consecutively numbered lines. These lines can be local lines connected to AMLC and ICS controllers, as well as NTS lines connected to an LTS.

Usually, you store SET_ASYNC commands in your PRIMOS.COMI file, so that all asynchronous lines are configured on your system at cold start. However, you can also use the SET_ASYNC command while the system is running to alter the characteristics of a particular line.

Prior to PRIMOS Rev. 21.0, the SET_ASYNC command was restricted for the System Administrator's use. Now, individual users can temporarily change the characteristics of their own line or any lines that they have assigned. These changes remain in affect until the user logs out or issues another SET_ASYNC command.

Certain options, such as USER_NUMBER, ASSIGNABLE, DISLOG, SPEED_DETECT, LOOP_LINE, and their converses, may affect buffer allocation or depend on hardware. These options are reserved for the System Administrator.

SET_ASYNC is a functional replacement, with extensions, for the AMLC command. It accepts data in the form of command line options that are straightforward and easy to understand. The SET_ASYNC command performs the following operations:

- Defines the protocol

- Enables Auto Speed Detect (ASD), if desired

- Sets line speed and parity

- Defines the initial terminal line characteristics

- Associates a user number with a line number or defines the line as assignable

- Allows the user to interactively change a line's characteristics

- Returns the line to its cold start configuration when the -SYSTEM option is specified

- Provides the default values shown in Table 4-2 with the -DEFAULT option

- Associates a set of identical line characteristics with a range of consecutively numbered lines

---

Caution

SET_ASYNC accepts decimal numbers only. Octal numbers are *not* supported.

Users also can issue the TERM command interactively at the keyboard or in a LOGIN.CPL file to specify full-duplex or half-duplex mode, disable the break character (CONTROL-P), and recognize XON/XOFF. For a detailed description of the TERM command, see the *PRIMOS Commands Reference Guide.*

The format of the SET_ASYNC command is

$$
\text{SET\_ASYNC} \left\{ \begin{array}{l} \text{-DISPLAY} \\ \text{-LINE n} \\ \text{-HELP} \end{array} \begin{array}{l} \text{[-LINE n]} \\ \text{[-TO m]} \quad \text{[options]} \end{array} \right\}
$$

The SET_ASYNC command can take the following arguments:

Argument   *Description*

-DISPLAY   Displays the current characteristics for the line.

-HELP   Displays the format of the command and a complete list of available options. The options that are restricted for the System Administrator's use are clearly marked.

-LINE n   Configures a line (or lines) with the selected options, where *n* is the required *decimal* line number or, when used with the -TO option, the first line number in a series configured with identical options. Valid line numbers range from 0 to 255 for local lines and from 1024 to 1279 NTS lines.

-TO m   Configures a range of consecutively numbered lines with identical options, where *m* specifies the last line number in a series beginning at the line number *n* given in -LINE. The value *m* must be greater than *n*.

There are many options to the SET_ASYNC command. An alphabetical list of options and valid abbreviations is provided in Table 4-1 for your convenience. A detailed description of each option follows.

TABLE 4-1.   SET_ASYNC Options and Abbreviations

Option	*Abbreviation*	*Option*	*Abbreviation*
-ASSIGNABLE *R*	-ASGN	-NO_ECHO	-NO_ECHO
-CHAR_LENGTH	-CL	-NO_ERROR_DETECTION	-NO_ERRDET
-DATA_SENSE_ENABLE	-DSE	-NO_LINE_FEED	-NOLF
-DATA_SET_CONTROL	-DSC	-NO_LOOP_LINE *R*	-NOLOOP
-DATA_SET_SENSE	-DSS	-NO_REVERSE_XOFF	-NOREVXOFF
-DEFAULT	-DEF	-NO_SPEED_DETECT *R*	-NOSD
-DISLOG *R*	-DISLOG	-NO_XOFF	-NO_XOFF
-DISPLAY	-DP	-PARITY	-PAR
-ECHO	-ECHO	-PROTOCOL	-PRO
-ERROR_DETECTION	-ERRDET	-REVERSE_XOFF	-REVXOFF
-HELP	-H	-SPEED	-SPEED
-LINE	-LINE	-SPEED_DETECT *R*	-SD
-LINE_FEED	-LF	-STOP_BITS	-SB
-LOOP_LINE *R*	-LOOP	-SYSTEM	-SYS
-NO_DATA_SENSE_ENABLE	-NODSE	-TO	-TO
-NO_DATA_SET_CONTROL	-NODSC	-USER_NUMBER *R*	-USER
-NO_DISLOG *R*	-NO_DISLOG	-XOFF	-XOFF

(*R*) = Restricted to the System Administrator

Note

These options replace the octal bitstrings used by the obsolete AMLC command. To find out the SET_ASYNC equivalent of a given AMLC command, you can use the interactive form of the CONVERT_AMLC_COMMANDS utility described on page 4-16.

The options to the SET_ASYNC command are as follows:

Option	*Description*
-SYSTEM	Sets the line characteristics to the cold start values contained in your PRIMOS.COMI file.
-DEFAULT	Sets all options to their default setting, shown in Table 4-2 on page 4-6, with the exception of any options further specified in the command line. For example the -DEFAULT option can be used to configure a line for a 9600 baud terminal:

```
SET_ASYNC -DEFAULT -LINE 78 -SPEED 9600
```

-ASSIGNABLE status	Indicates when the line is an assignable line, where *status* can be either YES or NO, the default. Assignable lines are required for many serial devices and some NTS applications. Only the System Administrator can change a line interactively from assignable to login and vice versa. The assignment applies only until the next system shutdown unless you edit your PRIMOS.COMI file to include the command line. The command SET_ASYNC -LINE n -ASGN NO is required to convert a previously assigned line to a regular login line. The user number for assigned lines must be zero. For more information, see Chapter 5, Assignable Asynchronous Lines.
-USER_NUMBER n	Associates a buffer with a physical line, where *n* is the decimal user number. This option is restricted. The default value is the line number specified in the -LINE option, plus two. The user number for assignable lines must be zero. Do not use this option with the -TO option unless you set a range of assignable lines to user number zero. For example,

```
SET_ASYNC -LINE 5 -TO 9 -ASGN YES -USER 0
```

-ECHO	Sets the line to be echoed by the host (full-duplex). This is the default.
-NO_ECHO	Sets the line so characters do not echo on the screen (half-duplex). -ECHO is the default.

*TABLE 4-2. Default Settings for Asynchronous Lines
Provided by the SET_ASYNC -DEFAULT Option*

Option	Setting
-ASSIGNABLE NO	Line is a login line.
-CHAR_LENGTH 8	Character length is 8 bits.
-DATA_SET_CONTROL	Enables modems and port selectors to recognize when information is being transmitted.
-ECHO	Full-duplex line.
-LINE_FEED	Echo both LINE FEED and RETURN for each RETURN.
-NO_DATA_SENSE_ENABLE	Data Set Sense is disabled.
-NO_DISLOG	Disables automatic disconnect.
-NO_ERROR_DETECTION	Input parity or input buffer overflow are not flagged with an ASCII NAK (negative acknowledgement) in the input buffer.
-NO_LOOP_LINE	Line is not in loopback mode.
-NO_REVERSE_XOFF	Reverse Flow Control is not enabled.
-NO_SPEED_DETECT	Disables Auto Speed Detect.
-PARITY NONE	Line parity disabled.
-PROTOCOL TTY	Line uses the terminal protocol.
-SPEED 1200[4]	Line speed is 1200 bits per second.
-STOP_BITS 1	One stop bit.
-USER_NUMBER (n+2)	Associates the buffers with the physical line number specified in the -LINE *n*. For example, the default buffer associated with -LINE 16 is buffer 18.
-XOFF	CONTROL-S stops and CONTROL-Q resumes the flow of data from the system to the terminal.

---

[4]Use the —SPEED option with the —DEFAULT option for terminals that operate at other speeds.

Option	*Description*
-PROTOCOL name	Defines the format and relative timing of data, where *name* may be any of the following:

> TTY   TTYUPC   TT8BIT TTY8
> TRAN TTYNOP   ASD   TTY8HS

For more information, see the sections Standard Communication Protocols on page 4-19, and Enabling Auto Speed Detect on page 4-24.

Do not use ASD or TTYNOP on an NTS line.

The obsolete protocols TTYHS, TRANHS, and TTYHUP are supported for older model 5054 AMLC controller boards.

-SPEED_DETECT	Enables Auto Speed Detect functionality to determine the speed of incoming data on the line. -SPEED_DETECT is equivalent to the protocol ASD. It is particularly useful for dialup lines. When you use this option, set the sample speed to -SPEED 9600. This option is restricted to the System Administrator. Do not use SPEED_DETECT on an NTS line.
-NO_SPEED_DETECT	Disables Auto Speed Detect. This option is restricted to the System Administrator.
-SPEED value	Sets the baud rate for the line, where *value* may be any of the following baud rates:

50	200	2400	19200	
75	300	3600	CLOCK	(2)
110	600	4800	SPEEDA	(3)
134.5	1200 (1)	7200	SPEEDB	(3)
150	1800	9600	SPEEDC	(3)

(1)   Supplied with the -DEFAULT option
(2)   Speed set with AMLCLK directive
(3)   Speeds set with the ASYNC JUMPER directive

For further information on the ASYNC JUMPER and AMLCLK directives, see Chapter 3, Useful Configuration Directives.

The maximum recommended speed for the last AMLC line is 300 baud. For more information see the section, AMLC I/O Interrupt Rate on page 6-4.

Option	*Description*
-STOP_BITS n	Signals the receiving device to wait for the next character, where n defines the number of stop bits to use, either 1 (default) or 2. All characters have 1 start bit, 1 parity bit, 7 information bits, and either 1 or 2 stop bits. Some devices that operate at slower speeds require 2 stop bits between characters. For this reason, an 11-bit character length and 2 stop bits are used for devices that operate at 110 baud.
-PARITY value	Sets the line parity to the desired setting or disables parity. *value* can be ODD, EVEN, or NONE (default). Parity checking is a common method of error detection. The transmitting device appends a parity bit to the end of each character based on the number of zeros or ones in the character. The receiving device calculates the parity as it reads the character, and, if the parity calculated agrees with the parity bit setting, accepts the character.
-DISLOG	Automatically logs out the user if the line is disconnected or the carrier signal is logically low. DISLOG can not be used for NTS lines and is restricted to the System Administrator. This option duplicates the functionality of the global configuration directive DISLOG YES for individual lines. For more information on the DISLOG directive, see Chapter 3, page 3-11.
-NO_DISLOG	Disables DISLOG. -NO_DISLOG is restricted to the System Administrator. The global configuration directive DISLOG YES overrides a per-line SET_ASYNC -NO_DISLOG command.
-CHAR_LENGTH n	Sets the number of information and parity bits per character. n can be 5, 6, 7, or 8 (default). Character length can be adjusted for the nonstandard character sizes required by Baudot terminals, telex lines, or foreign devices. PRIMOS right-justifies the bits in a byte and sets the leftmost bits to zero. The example below uses one stop bit.

Length	*Bit pattern*	
-CL 5	STOP 000XXXXX	START
-CL 6	STOP 00XXXXXX	START
-CL 7	STOP 0XXXXXXX	START
-CL 8	STOP XXXXXXXX	START

-LINE_FEED	Echos a line feed character for the RETURN key. This option is valid only when -NO_ECHO (half-duplex) is specified. This is the default.
-NO_LINE_FEED	Does not echo a line feed character for RETURN. -LINE_FEED is the default.

Option	Description
-XOFF	Recognizes CONTROL-Q and CONTROL-S (-XON and-XOFF) to stop and start the flow of data on the line from the host to the terminal. This is the SET_ASYNC default.
-NO_XOFF	Disables CONTROL-S and CONTROL-Q. Use this option for devices that cannot recognize these control key sequences. -XOFF is the default.
-REVERSE_XOFF	Enables Reverse Flow Control (RFC) for asynchronous lines. -NO_REVERSE_XOFF is the default. RFC sends XOFF characters to a device when the PRIMOS input ring buffer is 60% full. When the input ring buffer drops to 20% full, an XON character is sent to the device to indicate that transmission can resume.
	RFC also attempts to prevent DMQ input queue overruns for all ICS3 controllers and any ICS2 controllers that are not using BSC and ASYNC protocols.
	Choose this option only for lines connected to devices that can interpret XON and XOFF characters (such as PT45, PST100™, and PT200™ terminals).
-NO_REVERSE_XOFF	Disables Reverse Flow Control for the line. This is the default.
-DATA_SENSE_ENABLE	Enables the DATA_SET_SENSE protocol, also known as reverse channel protocol. Use this option for transmitting control information or for controlling the flow of data to devices that do not recognize XON/XOFF. NO_DATA_SENSE_ENABLE is the default.
-NO_DATA_SENSE_ENABLE	Disables the DSS protocol (reverse channel). This is the default.
-DATA_SET_SENSE	**ready_value** Supports devices that toggle an RS-232-C pin (usually pin 8) to indicate when they are busy/ready instead of using XON/XOFF. The DSS protocol sets *ready_value* as either HIGH (pin signal raised) or LOW (pin signal lowered). The default is HIGH. Some devices use pins other than pin 8. If this is the case, ask your Customer Support Center to arrange your cables so that the data set sense signal is wired into the pin used for carrier detect.
	If you use the -DSS, you must specify -NO_XOFF and use -DSE and -NO_DSE to enable and disable flow control.
	Data set sense is also referred to as buffered protocol or reverse channel protocol.

Option	Description
-DATA__SET__CONTROL	Required for modems and port selectors to recognize when a block of information is transmitted. This is the default. -DSC is ignored by terminals. Do not use -DSC on NTS lines.
-NO__DATA__SET__CONTROL	Disables the -DSC option.
-LOOP__LINE	Verifies when data is accurately transmitted. A software-enabled hardware loopback is used to send and return data on the same line. This option is used only by the System Administrator for testing.
-NO__LOOP__LINE	Disables the -LOOP__LINE option. This is the default. This option is restricted to the System Administrator.
-ERROR__DETECTION	This is used only for testing. When an input buffer overflows or when a parity error is detected, the incoming character is replaced with an ASCII 225, NAK. -NO__ERROR__DETECTION is the default.
-NO__ERROR__DETECTION	Prevents the line from sending an ASCII NAK character if an input parity or input buffer overflow error is sensed. This is the default.

Using SET__ASYNC

Here are some examples of common applications for the SET__ASYNC command.

Groups: The SET__ASYNC command line can configure a group of terminals attached to consecutive lines with a specific option or group of options. In the command line below, reverse flow control is enabled for ten terminals.

```
SET_ASYNC -LINE 129 -TO 138 -DEF -PRO TTY -SPEED 9600 -REVXOFF
```

Unused Lines: If you have a line or group of lines that are not connected to terminals, use the TTYNOP to prevent the CPU from wasting time attempting to interpret random electrical noise as commands. Use TTYNOP for any unused lines on a controller board. The following command configures five lines that are not currently connected to any devices.

```
SET_ASYNC -LINE 172 -TO 176 -PRO TTYNOP
```

Printers: TTYNOP is useful for printers, plotters, and other peripheral devices that do not carry on an interactive dialog with the CPU. The SET__ASYNC command shown below configures a line to a plotter.

```
SET_ASYNC -LINE 81 -PRO TTYNOP -SPEED 300
```

Modems: The SET__ASYNC command can configure dialup lines to automatically detect incoming baudrates and temporarily adjust the line's baudrate. This enables the dialup user to communicate with the Prime host. Modems also require DSC to recognize blocks of incoming data. If you have dialup lines or modems, include the DTRDRP, AMLTIM, LOUTQM, ASYNC JUMPER, and, optionally, DISLOG directives in your CONFIG file. In the example below one SET__ASYNC command sets ASD, DSC, a sample speed of 9600 baud, and a per-line DISLOG to log out the user if the line is disconnected.

```
SET_ASYNC -LINE 28 -DEF -SPEED_DETECT -SPEED 9600 -DSC -DISLOG
```

Internal Clocks: The last line of the last AMLC controller, in this example the eighth controller, sets the AMLC I/O interrupt rate for all lines attached to AMLC controllers. For more information on hardware defined clocks and timers, read the section, the AMLC I/O Interrupt Rate in Chapter 6. The following SET__ASYNC command configures the line to ignore any incoming signals and sets the timer to 10 interrupts per second.

```
SET_ASYNC -LINE 127 -PRO TTYNOP -SPEED 110 -SB 2
```

Assignable Lines: One SET__ASYNC command line can create a group of assignable lines. In this example, there are six assignable lines and the corresponding value of NAMLC is six. After issuing the ASSIGN command, the user can issue the SET__ASYNC command interactively to change the protocol or specify other options.

```
SET_ASYNC -LINE 194 -TO 199 -ASGN YES -PRO TRAN
```

Specific Devices: Issue individual SET__ASYNC commands for lines attached to devices that require special protocols, speeds, or options. The following command configures a line for an Arabic terminal.

```
SET_ASYNC -LINE 87 -PRO TT8BIT -PAR ODD -SPEED 9600
```

Some facilities use hard-copy terminals that cannot print lowercase characters. A SET__ASYNC command for such a device is shown below.

```
SET_ASYNC -LINE 17 -PRO TTYUPC -SPEED 110 -CL 7 -SB 2
```

The SET__ASYNC command can change flow control management from the standard XON/XOFF to DATA__SET__SENSE recognition for devices that toggle an RS-232-C pin. As shown in the command line below, DATA__SET__SENSE always requires DATA__SENSE__ENABLE and -NO__XOFF to operate.

```
SET_ASYNC -LINE 39 -DEF -DSE -DSS -NO_XOFF
```

Diagnostics: It is possible to run diagnostics on a line without changing any existing parameters. The following SET__ASYNC command issued from the supervisor terminal returns the data transmitted on line 71 back to the host and inserts a special character, an ASCII NAK, to flag errors.

```
SET_ASYNC -LINE 71 -LOOP -ERRDET
```

Making Interactive Changes: The SET_ASYNC -DISPLAY and -SYSTEM options are particularily useful when you change your line's configuration from your terminal. In the following example, the -DISPLAY option is used to show the line's current characteristics. The SET_ASYNC command is issued interactively to change the speed from 9600 baud to 1200 baud and set the line to full-duplex. The -DISPLAY option is used a second time to verify these changes.

```
OK, SET_ASYNC -DISPLAY

[Set_Async Rev 21.0 (c) Prime Computer, Inc. 1986]

LINE = 20
        PARity        = NONE      NO_ECHO
        PROtocol      = TTY       NO_XOFF
        SPEED         = 9600      NO_LOOP
        Stop_Bits     = 1         Line_Feed
        Char_Length   = 8         NO_Data_Sense_Enable
        REVerse_XOFF  = OFF       NO_ERRor_DETection
        ASsiGNable    = NO        NO_Speed_Detect
        Data_Sense_Set = LOW      NO_DISLOG
        Owner Process = 22        NO_Data_Set_Control

OK, SET_ASYNC -ECHO -SPEED 1200
OK, SET_ASYNC -DISPLAY

[Set_Async Rev 21.0 (c) Prime Computer, Inc. 1986]

LINE = 20
        PARity        = NONE      ECHO
        PROtocol      = TTY       NO_XOFF
        SPEED         = 1200      NO_LOOP
        Stop_Bits     = 1         Line_Feed
        Char_Length   = 8         NO_Data_Sense_Enable
        REVerse_XOFF  = OFF       NO_ERRor_DETection
        ASsiGNable    = NO        NO_Speed_Detect
        Data_Sense_Set = LOW      NO_DISLOG
        Owner Process = 22        NO_Data_Set_Control
```

Use the -SYSTEM option to return the line to its cold start values when you are finished. Optionally, you can issue the SET_ASYNC -DISPLAY once again to verify this.

```
OK, SET_ASYNC -SYSTEM
OK, SET_ASYNC -DISPLAY

[Set_Async Rev 21.0 (c) Prime Computer, Inc. 1986]

LINE = 20
        PARity        = NONE      NO_ECHO
        PROtocol      = TTY       NO_XOFF
        SPEED         = 9600      NO_LOOP
        Stop_Bits     = 1         Line_Feed
        Char_Length   = 8         NO_Data_Sense_Enable
        REVerse_XOFF  = OFF       NO_ERRor_DETection
        ASsiGNable    = NO        NO_Speed_Detect
        Data_Sense_Set = LOW      NO_DISLOG
        Owner Process = 22        NO_Data_Set_Control
```

SET__ASYNC Error Messages

The SET__ASYNC command verifies command line options before it defines or changes an asynchronous line's characteristics. Typical errors involve contradictory, ambiguous, or incorrect values in the command line. If an error is detected, the command is aborted and no action is taken. One of the error messages below is displayed and you are returned to command level. However, if any error occurs that is termed WARNING in the error messages below, the command is processed.

All error messages display the line number that you specified, if possible. This is especially helpful when you specify multiple line numbers.

Bad buffer number. (SET_ASYNC)
> The -USER__NUMBER option specifed an invalid or out-of-range buffer number.

Bad line number. (SET_ASYNC)
> An invalid or out-of-range line number was specified.

Bad protocol. (SET_ASYNC)
> An invalid or unsupported protocol was specified.

Buffer in use. (SET_ASYNC)
> An attempt was made to specify a buffer that is currently assigned to and in active use by another user or process.

—CHAR_LENGTH 8 must be specified with a —PARITY value of NONE (line <n>). (SET_ASYNC)
> The user specified odd or even parity with a character length of 8 bits. PRIMOS uses the parity (eighth) bit for data.

-DATA_SENSE_ENABLE cannot be specified because —DISLOG was previously set (line <n>). (SET_ASYNC)
> An attempt was made to specify DSE for a line using DISLOG. DISLOG interprets some DSE signals as logically low signals and disconnects the line.

—DISLOG cannot be specified because —DATA_SENSE_ENABLE was previously set (line <n>). (SET_ASYNC)
> An attempt was made to specify -DISLOG for a line using DSE. DSE translates logically low signals into XONs and XOFFs and cannot recognize a DISLOG disconnect.

Incompatible USER_NUMBER and ASSIGNABLE options specified (line <n>). (SET_ASYNC)
> The user number zero was associated with a login line *n* or a nonzero user number was associated with an assignable line *n*.

Invalid ASD use. (SET_ASYNC)
> The ASD protocol was specified for an NTS line. NTS hardware contains similar functionality and all NTS lines are configured to detect and adjust transmission speeds.

Invalid <argument> value <x> (line <n>). (SET_ASYNC)
> An invalid, unrecognized, or missing argument value *x* was specified for *argument* on line *n*.

`Invalid line number(s) specified.  (SET_ASYNC)`
> The line number or -TO number specified was out of range or otherwise invalid.

`Invalid protocol value <protocol name> (line <n>).  (SET_ASYNC)`
> The *protocol name* entered is not one of the expected values. See the section, Standard Communication Protocols, on page 4-19 for a detailed description of the protocols currently supported.

`Invalid sample speed for ASD <speed>.  (SET_ASYNC)`
> The baud rate specified for the line *speed* was not one of the following recognized ASD values: 110, 300, 600, 1200, 2400, 4800, 9600, or 19200. Read the description of valid ASD speed on page 4-21 for more information.

`Invalid speed value <baud rate> (line <n>).  (SET_ASYNC)`
> The *baud rate* entered was not one of the expected values. See the -SPEED option described on page 4-7 for a list of valid speeds.

`Invalid use of assign line buffer.  (SET_ASYNC)`
> An attempt was made to specify for a login line a buffer reserved for assignable lines.

`Invalid use of DISLOG.  (SET_ASYNC)`
> An attempt was made to specify DISLOG for a line using DSE. DISLOG interprets some DSE signals as logically low and disconnects the line.

`Invalid use of remote buffer.  (SET_ASYNC)`
> An attempt was made to specify a buffer reserved for PRIMENET remote users for an assignable, login, or NTS line.

`Line in use.  (SET_ASYNC)`
> A user issued a SET__ASYNC command for a line that is actively in use by another user or process.

`Line is assigned by another user (line <n>).  (SET_ASYNC)`
> A user issued a SET__ASYNC command for a line that is assigned by another user.

`Line is not assignable (line <n>).  (SET_ASYNC)`
> A user issued a SET__ASYNC -LINE *n* -ASGN YES command for a login line. Only the System Administrator can change a login line to an assignable line and vice versa.

`Line has not been assigned (line <n>).  (SET_ASYNC)`
> A user issued a SET__ASYNC -LINE *n* -ASGN NO command for a login line. Only the System Administrator can change a login line to an assignable line and vice versa.

`Line not owned by you.  (SET_ASYNC)`
> A user issued a SET__ASYNC command for a line currently assigned or actively used by some another user or process.

`Line number MUST be specified.  (SET_ASYNC)`
> The SET__ASYNC command was issued from the supervisor terminal without the required line number.

`Line number specified with the -TO option must be greater than the line number specified`
`    with the -LINE option.  (SET_ASYNC)`

The -TO line number was less than or equal to the -LINE argument.

`No NTS available.  (SET_ASYNC)`

An attempt was made to configure an NTS line before NTS is running.

`-<option> and -NO_<option> cannot both be specified at the same time (line <n>).`
`(SET_ASYNC)`

An option and its converse were specified for line n at the same time. For example, specifying both -XOFF and -NO__XOFF for the same line produces this error message.

`Speed value <baud rate> can not be set (line <n>).  (SET_ASYNC)`

The *baud rate* selected for line n is valid but is not available in the current physical terminal configuration.

`The <option name> option is not supported on NTS lines (line <n>).  (SET_ASYNC)`

One or more of the following options were specified for an NTS line n: -SPEED__DETECT, -NO__SPEED__DETECT, -DISLOG, -NO__DISLOG, -DATA__SET__CONTROL, or -NO__DATA__SET__CONTROL.

`The <protocol name> protocol is not supported on NTS lines (line <n>).  (SET_ASYNC)`

The TTYNOP or ASD protocol was specified for an NTS line n.

`Unknown return code detected (line <n>).  (SET_ASYNC)`

An unexpected error prevented SET__ASYNC from setting a line's characteristics. If subsequent attempts to issue the command fail, contact your Customer Support Center.

`Unexpected error in CL$PIX.  (SET_ASYNC)`

An unexpected error prevented SET__ASYNC from returning an error message. If subsequent attempts to issue the command fail, contact your Customer Support Center.

`You cannot refer to an NTS line when NTS is not started (line <n>).  (SET_ASYNC)`

This message is displayed when line n is an NTS line and NTS has not been started.

`Warning: LINE_FEED option not meaningful unless -NO_ECHO is specified (line <n>).`
`(SET_ASYNC)`

This message is intended to notify the operator that -LINE__FEED was specified for line n, -LINE__FEED is only meaningful if the line is half-duplex. This combination is ignored.

`Warning: ECHO option not meaningful unless LINE_FEED is specified (line <n>).  (SET_ASYNC)`

This message is intended to notify the operator that ECHO was specified for line n, -LINE__FEED is only meaningful if the line is half-duplex. This combination is ignored.

`Warning: Range specified crosses invalid lines. Lines <n> to <m> have been skipped.`
`(SET_ASYNC)`

The SET__ASYNC -LINE n -TO m option included line numbers between 255 and 1024.

The CONVERT_AMLC_COMMANDS Utility

CONVERT_AMLC_COMMANDS is a utility that translates AMLC commands to their equivalent SET_ASYNC commands. By using this utility, you can easily translate the AMLC commands in your existing PRIMOS startup command input file without altering the other commands or corrupting your original file. Interactive mode is particularly useful when you are learning how to use SET_ASYNC.

CONVERT_AMLC_COMMANDS is not an installed PRIMOS command. It is supplied in the directory TOOLS as CONVERT_AMLC_COMMANDS.RUN. If you wish, you can copy this file into the directory CMDNC0 and invoke it as a regular PRIMOS command. Otherwise, use the following format to invoke this utility:

<u>R</u>ESUME TOOLS>CONVERT_AMLC_COMMANDS $\left\{ \begin{array}{l} \text{input-file output-file} \\ \text{-INTERACTIVE} \\ \text{-HELP} \end{array} \right\}$

Option	*Description*
input-file output-file	Translates AMLC commands contained in the input file to their SET_ASYNC equivalent, and stores them in the output file. Other commands are written unchanged to the output file. The input file can be your current PRIMOS.COMI, or any other file that contains AMLC commands. The output file is created by this utility. Do not choose the input file or any other existing file as your output file.
<u>-INTER</u>ACTIVE	Queries you for an AMLC command, translates the command, and displays the SET_ASYNC equivalent on the screen. Type QUIT to exit from interactive mode and return to PRIMOS.
<u>-HELP</u>	Shows the syntax of the command line.

Note

CONVERT_AMLC_COMMANDS generates a separate SET_ASYNC command for each AMLC command it processes. This utility cannot recognize when consecutive lines have identical directives and cannot recognize a logical group of AMLC commands and combine them into a single SET_ASYNC command.

The following message is displayed when you invoke the -HELP option.

```
OK, RESUME TOOLS>CONVERT_AMLC_COMMANDS -HELP

[Convert_Amlc_Commands  Rev 20.2 (c) Prime Computer 1985]
USAGE:
CONVERT_AMLC_COMMANDS  { inputpathname outputpathname | -Help | -INTERactive }
```

The following example demonstrates interactive mode:

```
OK, RESUME TOOLS>CONVERT_AMLC_COMMANDS -INTER
[Convert_Amlc_Commands  Rev 20.2 (c) Prime Computer 1985]

>AMLC 4 TTY 2413 020006

SET_ASYNC -LINE 4 -PRO TTY -DEFAULT -SPEED CLOCK -LF -BUFFER 6

>AMLC 5 TTY 2413 020007

SET_ASYNC -LINE 5 -PRO TTY -DEFAULT -SPEED CLOCK -LF -BUFFER 7

>AMLC 7 TTY 2213 020020

SET_ASYNC -LINE 7 -PRO TTY -DEFAULT -SPEED 300 -LF -BUFFER 16

>AMLC 10 TRAN 2313 020000

SET_ASYNC -LINE 8 -PRO TRAN -DEFAULT -LF

>AMLC 11 TRAN 2313

SET_ASYNC -LINE 9 -PRO TRAN -DSC -NO_LOOP -SPEED 1200 -NO_REV_XOFF
-SB 1 -PAR NONE -CL 8

> AMLC TTY 61 2413 0

SET_ASYNC -LINE 49 -PRO TTY -DEFAULT -SPEED CLOCK -LF -NO_XOFF

> AMLC TTY 137 2713 20000

SET_ASYNC -LINE 95 -PRO TTY -DEFAULT -SPEED 1800 -LF

> AMLC TTYNOP 106 2713 0

SET_ASYNC -LINE 70 -PRO TTYNOP -DEFAULT -SPEED 1800 -LF -NO_XOFF

> AMLC 77 0213 0

SET_ASYNC -LINE 63 -PRO TRAN -DEFAULT -NO_DSC -SPEED 300 -LF -NO_XOFF

> AMLC 111 2313 2000

SET_ASYNC -LINE 73 -PRO TRAN -DEFAULT -LF -NO_XOFF -DSS LOW

> AMLC 65 TTY 2213 0

SET_ASYNC -LINE 53 -PRO TTY -DEFAULT -SPEED 300 -LF -NO_XOFF

>QUIT

OK,
```

In the following example the CONVERT_AMLC_COMMANDS utility is given an input file called PRIMOS.COMI. An abbreviated listing of this file is provided.

```
CONFIG -DATA CONFIG              /* Name of config file for start-up
COMO PRIMOS.COMO                 /* Open como file to record this startup
ADDISK 3660 14061 3462 71261     /* Add partitions to the system
COMO -NTTY                       /* Turn off echo to system console
AMLC  TTY     0  2413  20002     /* User terminal set to 9600 baud
AMLC  TTY     1  2413  20003     /* User terminal set to 9600 baud
AMLC  TTY     2  2413  20004     /* User terminal set to 9600 baud
AMLC  TTY     3  2413  20005     /* User terminal set to 9600 baud
AMLC  TTY     4  2413  20006     /* Set TTY Protocol and 9600 baud
AMLC  ASD     4                  /* Enable ASD, Retain TTY and baud
AMLC  TTY     5  2313  20007     /* Dial-in line set to 1200 baud
AMLC  TTY     6  2413  20010     /* ONTEL line set at 9600 baud
AMLC  TTY     7  2413  20011     /* OAS terminal set to 9600 baud
AMLC  TTY    10  2413  20012     /* Forms terminal set to 9600 baud
AMLC  TTY    11  2413  20013     /* Unused line
AMLC  TTY    12  2413  20014     /* Unused line
AMLC  TTY    13  2413  20015     /* Unused line
AMLC  TTYNOP 14  2413  20000     /* Laser Printer set at 9600 baud
AMLC  TRAN   15  2413  20000     /* Assigned Line for 9600 baud Printer
AMLC  TTYNOP 16  2313  20000     /* OASTAT Printer set at 1200 baud
AMLC  TTYNOP 17  2213  20000     /* Letter Quality Printer set at 300 baud
/*
CO SYSTEM>DSM.SHARE.COMI 7           /* Install DSM database
CO SYSTEM>BASICV.SHARE.COMI 7        /* Share BASICV
CO SYSTEM>DBG.SHARE.COMI 7           /* Share DBG
CO SYSTEM>EMACS.SHARE.COMI 7         /* Share EMACS
CO SYSTEM>FORMS.SHARE.COMI 7         /* Share FORMS
CO SYSTEM>MIDASPLUS.SHARE.COMI 7     /* Share MIDASPLUS
CO SYSTEM>OAS.SHARE.COMI 7           /* Share OAS
CO SYSTEM>INFORMATION.SHARE.COMI 7/* Share Prime INFORMATION
CLOSE 7                          /* Close file unit 7
SHARE SYSTEM>ED2000 2000         /* Share the Editor
SHARE SYSTEM>S2050 2050 700
R SYSTEM>S4000 1/1
SHARE 2050
R SYSTEM>SP4000 1/10
SHARE SYSTEM>S$2167 2167         /* Share Spool libraries
R SYSTEM>S$4000 1/12
PROP PRO -START                  /* Start printer PRO
START_DSM
START_NET
START_NTS
/*
COMO -END -TTY                   /* Close como file, turn on echo
CO -END                          /* End of Primos.comi startup
```

The source filename, PRIMOS.COMI, and the target filename, CONVERT.COMI, are supplied in the command line, as shown below.

```
OK, RESUME TOOLS>CONVERT_AMLC_COMMANDS PRIMOS.COMI CONVERT.COMI
[Convert_Amlc_Commands  Rev 20.2 (c) Prime Computer 1985]
OK,
```

CONVERT_AMLC_COMMANDS processes the input file, creates the output file in your current directory, and returns to PRIMOS. When you list the contents of your directory with an LD command, the output filename appears. Here is an abbreviated listing of the target file CONVERT.COMI. Notice that CONVERT_AMLC_COMMANDS has translated only AMLC commands. All the other commands are the same as before.

```
CONFIG -DATA CONFIG                        /* Name of config file for start-up
COMO PRIMOS.COMO                           /* Open como file to record this startup
ADDISK 3660 14061 3462 71261               /* Add partitions to the system
COMO -NTTY                                 /* Turn off echo to system console
SET_ASYNC -LINE 0  -PRO TTY    -DEFAULT -SPEED CLOCK -LF -USER 2    /* User termin
SET_ASYNC -LINE 1  -PRO TTY    -DEFAULT -SPEED CLOCK -LF -USER 3    /* User termin
SET_ASYNC -LINE 2  -PRO TTY    -DEFAULT -SPEED CLOCK -LF -USER 4    /* User termin
SET_ASYNC -LINE 3  -PRO TTY    -DEFAULT -SPEED CLOCK -LF -USER 5    /* User termin
SET_ASYNC -LINE 4  -PRO TTY    -DEFAULT -SPEED CLOCK -LF -USER 6    /* Set TTY Pro
SET_ASYNC -LINE 4  -PRO ASD                                        /* Enable ASD,
SET_ASYNC -LINE 5  -PRO TTY    -DEFAULT -LF -USER 7                 /* Dial-in lin
SET_ASYNC -LINE 6  -PRO TTY    -DEFAULT -SPEED CLOCK -LF -USER 8    /* ONTEL line
SET_ASYNC -LINE 7  -PRO TTY    -DEFAULT -SPEED CLOCK -LF -USER 9    /* OAS termina
SET_ASYNC -LINE 8  -PRO TTY    -DEFAULT -SPEED CLOCK -LF -USER 10   /* Forms termi
SET_ASYNC -LINE 9  -PRO TTY    -DEFAULT -SPEED CLOCK -LF -USER 11   /* Unused line
SET_ASYNC -LINE 10 -PRO TTY    -DEFAULT -SPEED CLOCK -LF -USER 12   /* Unused line
SET_ASYNC -LINE 11 -PRO TTY    -DEFAULT -SPEED CLOCK -LF -USER 13   /* Unused line
SET_ASYNC -LINE 12 -PRO TTYNOP -DEFAULT -SPEED CLOCK -LF            /* Laser Print
SET_ASYNC -LINE 13 -PRO TRAN   -DEFAULT -SPEED CLOCK -LF            /*Assigned Lin
SET_ASYNC -LINE 14 -PRO TTYNOP -DEFAULT -LF                         /* OASTAT Prin
SET_ASYNC -LINE 15 -PRO TTYNOP -DEFAULT -SPEED 300 -LF              /* Letter Qual
/*
CO SYSTEM>DSM.SHARE.COMI 7              /* Install DMS database
CO SYSTEM>BASICV.SHARE.COMI 7           /* Share BASICV
CO SYSTEM>DBG.SHARE.COMI 7              /* Share DBG
CO SYSTEM>EMACS.SHARE.COMI 7            /* Share EMACS
CO SYSTEM>FORMS.SHARE.COMI 7            /* Share FORMS
CO SYSTEM>MIDASPLUS.SHARE.COMI 7        /* Share MIDASPLUS
CO SYSTEM>OAS.SHARE.COMI 7              /* Share OAS
CO SYSTEM>INFORMATION.SHARE.COMI 7/* Share Prime INFORMATION
CLOSE 7                                 /* Close file unit 7
SHARE SYSTEM>ED2000 2000                /* Share the Editor
SHARE SYSTEM>S2050 2050 700
R SYSTEM>S4000 1/1
SHARE 2050
R SYSTEM>SP4000 1/10
SHARE SYSTEM>S$2167 2167                /* Share Spool libraries
R SYSTEM>S$4000 1/12
PROP PRO -START                         /* Start printer PRO
START_DSM
START_NET
START_NTS
COMO -END -TTY                          /* Close como file, turn on echo
CO -END                                 /* End of Primos.comi startup
```

Standard Communication Protocols

Standard communication protocols allow the orderly transfer of data. These protocols usually define the format and relative timing of the information exchanged for a specific device. SET_ASYNC supports several asynchronous protocols that govern the actions taken by the PRIMOS Asynchronous Device Interface Modules (ASYNC DIMs). The acceptable values for the SET_ASYNC -PROTOCOL option are TTY (the default), TRAN, TT8BIT, TTYUPC, TTYNOP, TTY8, TTY8HS, and ASD. NTS lines do not support ASD and TTYNOP. The basis for selection of the protocol is discussed below.

Protocols for older model AMLC boards (model 5054) are discussed in Appendix A, Obsolete and Rarely Used Commands, on page A-4.

TRAN: TRAN, the transparent protocol, is often used by lines connected to peripheral devices or to other computers. Choose TRAN when it is not necessary to echo input, convert *carriage returns* to *line feeds*, or specifically acknowledge *carriage returns* and *line feeds*. The key sequence CONTROL-P has no special meaning under this protocol and is passed through to the program.   All characters pass as data unless XON/XOFF flow control is set.

TTY: TTY, the terminal protocol, is the protocol assigned at cold start to lines controlling interactive terminals.   When the TTY protocol is chosen, all input from the terminal is echoed if the line is set for full-duplex; both a *carriage return* and a *line feed* are echoed following each *carriage return.*   The high order bit (the ASCII code parity bit) of each character input from the terminal is forced on. CONTROL-P and BREAK are interpreted as a QUIT command if the terminal is connected to the system as a user terminal.

If no protocol is given in the SET_ASYNC command line, TTY is assigned by the operating system.

If the line is an assignable line, CONTROL-P, BREAK, and *line feed* input from the terminal are ignored and discarded.   A *carriage return* entered at the terminal is transmitted as a new line (or line feed) to the program requesting input.

Most terminals use this protocol and separate SET_ASYNC commands can be issued for each of your terminals.   However, when terminals are attached to a series of consecutive lines, this protocol can be assigned to the entire group with only one command line as follows:

```
SET_ASYNC -LINE 14 -TO 35 -DEF -PRO TTY -SPEED 9600
```

TT8BIT: TT8BIT behaves in the same manner as the TTY protocol except that the high order bit (ASCII parity bit) is not forced on for each character entered at the terminal. All control characters are handled in the same manner as the TTY protocol.

This protocol may not be used for remote users and is recommended only for local Arabic DM5E/PLUS terminals.

To use this protocol, you must set the line parity to ODD, as shown in the example below:

```
SET_ASYNC -LINE 5 -PRO TT8BIT -PAR ODD -SPEED 9600
/* Line 5: odd parity enabled, 9600 baud
```

TTYUPC: TTYUPC, the uppercase translating protocol, avoids sending lowercase output to terminals or peripheral devices that cannot print lowercase characters.   This is the only difference between TTY and TTYUPC protocols.

TTYNOP: TTYNOP instructs the asynchronous DIM to ignore all traffic on the line.   If you have a line that is not connected to a terminal, use this protocol to avoid wasting CPU time in attempts to interpret random noise on this line as valid commands.   NTS lines do not support TTYNOP.

```
SET_ASYNC -LINE 51 -PRO TTYNOP
```

TTY8 and TTY8HS: TTY8 adapts the standard terminal protocol, TTY, for devices that use Prime ECS.   If you have model number 5052 or 5054 AMLC controller boards, use TTY8HS, a functional equivalent of TTY8 specifically designed for this hardware.

ASD: ASD   allows PRIMOS to detect and automatically adjust the line's speed to agree with an incoming signal. Set the line's speed to 9600 baud and enable ASD. Depending on how you configure the line, PRIMOS can detect baud rates of 110, 300, 600, 1200, 2400, 4800, 9600, and 19200 bits per second.   Use ASD on login lines only. The system rejects any attempt to issue the ASD protocol for an assignable line or an NTS line.   ASD can also be enabled with the SET_ASYNC option -SPEED_DETECT. For further information, read the section, Enabling Auto Speed Detect, on page 4-24.

Configuring Asynchronous Lines

Use this procedure to configure asynchronous lines. The parameters take effect the next time you cold start your system.

1. Calculate the values for the following directives:

   > NTUSR (terminal users plus supervisor terminal)
   > NAMLC (assignable lines)
   > NTSUSR (NTS terminal users)
   > NTSASL (NTS assignable lines)

   For instructions on calculating these values, see Chapter 3, Useful Configuration Directives, and *The System Administrator's Guide, Volume 1: System Configuration.*

2. Use the octal conversion function of the TYPE command, as shown below, to display the octal equivalent of the decimal values.

   ```
   TYPE [TO_OCTAL 64]
   100
   ```

3. Use EMACS or ED to include these values in your CONFIG file.

4. Optionally, use the AMLBUF directive to change buffer sizes to fine tune the system and increase performance.   This directive is described in Chapter 6, Allocating Buffers.

5. Optionally, use the NTSBUF and NTSABF directives to set buffer sizes for NTS lines. These directives are described in Chapter 3, Useful Configuration Directives.

6. Use the SET_ASYNC command in your PRIMOS.COMI file to define assignability, protocols, baud rates, and other parameters for asynchronous lines.

Determining Line Numbers

A maximum of 256 asynchronous communication lines can be attached to your system, including one line for the supervisor terminal. These lines are attached to your AMLC, ICS, and LTS controllers and are identified by number. Local line numbers begin at 0 and cannot exceed 255. NTS line numbers start at 1024 and cannot exceed 1279.

Determine the decimal line number required to reconfigure an asynchronous line with SET_ASYNC by issuing either the LOGIN, LOGOUT, or STATUS USERS commands. These methods apply when the default line number assignments are used. If you cannot use these methods to determine a line number, use the procedure shown in Appendix B, Determining Physical Line Numbers.

Using LOGIN and LOGOUT

You can generally determine the physical line number from the user number that is displayed after login or logout by using the following formula:

Physical Line Number = User Number - 2

The following examples show how to apply this formula.

```
Login please.
LOGIN TINMAN
Password? ░░░░░░

TINMAN (User 11) Logged in Friday, 28 Aug 87 14:01:20.
Welcome to PRIMOS version 21.0
Copyright (c) Prime Computer, Inc. 1987.
Last Login Thursday, 27 Aug 87 10:54:40.
```

In this example, TINMAN is User 11 on line number 9 (11 - 2 = 9). Zero is the first valid line number on all Prime systems. In the following example DOROTHY is User 2 on line number 0 (2 - 2 = 0).

```
OK, LO

DOROTHY (User 2) Logged out Tuesday, 25 Aug 87 17:59:12.
Time used:  08h 41m connect, 04m 20s CPU, 01m 06s I/O.
```

Using STATUS USERS

The STATUS USERS command, as shown below, displays the user ID, the user number in decimal, the line number in octal and decimal, and the devices currently in use.

In the example show on page 4-23, W.WITCH, user number 40, is attached to physical line number 38. G.WITCH, user number 65, is a remote user and is assigned a virtual line from a pool reserved for remote users with the NRUSR directive. This is indicated by the abbreviation *rem* in the line number field.

OK, <u>STATUS USERS</u>

```
                    Line
User            No   oct(  dec)  Devices
SYSTEM           1      asr      <OZSYS1> AL77
DOROTHY          2     0(    0)  <MUNCH8> <KANSAS> <OZSYS1>
TOTO             6     4(    4)  <MUNCH8> <KANSAS> <OZSYS1>
RAINBOW          7     5(    5)  <KANSAS> <MUNCH8> <OZSYS1>
TINMAN          11    11(    9)  <OZSYS1>
SCARECROW       14    14(   12)  <OZSYS1> <KANSAS>
WIZARD          17    17(   15)  <OZSYS1>
Y.B.ROAD        22    24(   20)  <MUNCH8> <EMCITY>
AUNTI.EM        24    26(   22)  <KANSAS> (to BARN)
UNCLE.H         25    27(   23)  <KANSAS> (to BARN)
LION            30    34(   28)  <FOREST> <OZSYS1> <CASTLE>
WIZARD          38    44(   36)  <OZSYS1> <EMCITY>
W.WITCH         40    46(   38)  <BROOM1> <OZSYS1> <CASTLE>
MONKEY          41    slave      <CASTLE>
BALLOON         43    slave      <EMCITY>
G.WITCH         65     rem       <MUNCH8> (from NORTHl)
F.BAUM          66  2000( 1024)  <OZSYS1>
A.FREID         67  2001( 1025)  <OZSYS1>
LB.MAYER        68  2002( 1026)  <OZSYS1>
KIP.Y           69  2003( 1027)  <OZSYS1>
H.ARLEN         70  2004( 1028)  <OZSYS1>
WIZARD          88    phant      <EMCITY>
YTSMAN          91    phant      <OZSYS1>
WIZARD          93    phant      <EMCITY> <OZSYS1> AL66
LOGIN_SERVER    94     LSr       <OZSYS1> (3)
NTS_SERVER      95     ncm       <OZSYS1>
TIMER_PROCESS   96    kernel     <OZSYS1>
NETMAN          98     nsp       <OZSYS1>
TAPE_PHANTOM   100    phant      <OZSYS1>
LQ.PRINTER     101    phant      <OZSYS1> PR0
BATCH_SERVICE  103    phant      <OZSYS1> (2)
DSMSR          105    phant      <OZSYS1>
DSMASR         106    phant      <OZSYS1>
DSMNETSR       107    PHANT      <OZSYS1>
SYSTEM_MANAGER 108     dsm       <OZSYS1>
PRINT_SERVER   111    phant      <OZSYS1>
NM_SERVER      114    phant      <OZSYS1>
RAINBOW        117    child      <KANSAS> <MUNCH8> <OZSYS1>
WIZARD         119    batch      <EMCITY> <OZSYS1>
```

WIZARD, user numbers 17, 38, 88, 93, and 119 is logged in at two devices. Although this user name appears five times in this display, WIZARD is attached to only two physical line numbers (lines 15 and 36). WIZARD, user number 88, is a phantom process. It is not associated with a terminal and does not require a line. WIZARD, user number 93, is assigned line 54 indicated by the characters AL66. WIZARD, user number 119, is a batch process.

The STATUS USERS command displays NTS users after PRIMENET remote login users. NTS line numbers start at 2000_8 (1024). There are five NTS users in the preceding example.

The supervisor terminal is connected to the Virtual Control Panel on the CPU and uses a separate line that is configured with the ASRATE directive. This is indicated by the abbreviation *asr* in the line number field.

Enabling Auto Speed Detect

Auto Speed Detect (ASD) is particularily useful if your system has dialup lines. Configure dialup lines with ASD and a sample speed of 9600 baud. A user who makes the telephone connection presses the carriage return key a few times. PRIMOS uses these carriage returns to determine the device's baud rate. If the baud rate is one of the supported SET_ASYNC speeds or a hard-wired ASYNC JUMPER speed, PRIMOS changes the rate at which it sends data out over the line to match the device's baud rate.

After PRIMOS has determined the line speed, the LOGIN PLEASE message is displayed and the user is allowed to log in. When the user logs out, or after a forced logout, the line returns to ASD. To disable ASD and set specific speed, issue a SET_ASYNC -LINE *n* -NO_SPEED_DETECT -SPEED *value* command.

Do not specify ASD for an NTS line. NTS hardware contains similar functionality and all NTS line are automatically configured to detect and adjust transmission speeds.

Interactive users can not specify the -PRO ASD or -SPEED_DETECT options for the SET_ASYNC command. These are privileged options that are restricted for the System Administrator's use.

There are two methods of enabling ASD. The SET_ASYNC command allows you to specify a protocol and ASD in one command line:

```
SET_ASYNC -LINE 248 -PRO TTY -SPEED_DETECT -SPEED 9600
```

ASD can also be enabled with two command lines. First, configure the line as you normally would using the SET_ASYNC -PROTOCOL option to specify the appropriate protocol for the line. Then explicitly set ASD as an alternate protocol for the line with another SET_ASYNC command.

```
SET_ASYNC -LINE 5 -PRO TTY -SPEED 9600
SET_ASYNC -LINE 5 -PRO ASD
```

Note

When the system enables ASD, it reconfigures the line for character length 7, odd parity, and 1 stop bit, to prepare for the next user. If it is necessary, use the SET_ASYNC command interactively to change these line characteristics after you establish the connection. Interactive changes to line characteristics are in effect for the current session only; when you log out, ASD is reenabled.

The three configuration directives that affect ASD are ASYNC JUMPER, DISLOG, and DTRDRP. Use these directives in the configuration file as follows:

● Specify the valid jumper speeds with ASYNC JUMPER.

● Set DISLOG systemwide or for the specific line. This directive logs out the user if the user disconnects without logging out.

● Include DTRDRP. This directive drops the DTR signal and returns the line to ASD when the user logs out or is logged out because of DISLOG.

Note

If you do not have the DTRDRP and DISLOG directives in your configuration file, set the AMLTIM *disctime* and *gracetime* parameters to values greater than 0.

How you use the ASYNC JUMPER directive depends on whether your system uses AMLC boards, ICS controllers, or both.

● For AMLC boards, the ASYNC JUMPER directive tells PRIMOS how the hardware jumpers are set. To set the jumpers to other non-default speeds, have a representative from your Customer Support Center change the hardware settings. If you do not use the ASYNC JUMPER directive with AMLC lines, only baud rates of 110, 300, 1200, and 9600 are detected.

● All AMLC controllers must be jumpered identically for ASD to detect all speeds on every AMLC controller.

● For ICS controllers, the ASYNC JUMPER directive assigns the appropriate speeds to the lines.

● For a mixture of both AMLC boards and ICS controllers, the speeds in the ASYNC JUMPER directive must match the hardware-jumpered speeds on the AMLC board.

For further details on these directives, see Chapter 3, Useful Configuration Directives, and the *System Administrator's Guide, Volume I: System Configuration.*

ASSIGNABLE ASYNCHRONOUS LINES

Assignable asynchronous lines are a class of communication lines that are used by programs or devices rather than interactive terminal users. The first part of this chapter shows how to assign, configure, and release assignable lines with the ASSIGN ASYNC, SET_ASYNC, and UNASSIGN ASYNC commands. The second part discusses the NTS_ASSOCIATE, NTS_UNASSOCIATE, and NTS_LIST_ASSOCIATE commands. These commands allow NTS users to define, release, and display the active paths between devices.

Serial Devices

Serial devices process a series of consecutive parts, such as the bits of a character or the characters of a word, to form a whole. Typical serial devices are certain printers, and certain hard-copy terminals when these terminals are used as printers. The following Prime products require assignable asynchronous lines to communicate with the host computer.

- 30/60 cps hard-copy terminal (Product Number 3115F)

- 300 lpm, matrix lineprinter/plotter (Product Number 3126F)

- 200 cps bidirectional matrix printer (Product Number 3350F)

- 200 cps bidirectional hard-copy terminal (Product Number 3351F)

- 55 cps bidirectional letter quality printer (Product Number 3185)

- 600 lpm, 96 character matrix printer (Product Number 3226F)

- High-performance PRIME MEDUSA™ work station (Product Number PW 95)

- 10 ppm laser page printer (Product Number 3410)

Prime supports several other serial devices, such as plotters, card readers and punches, microcomputers, optical scanning devices, and other compatible devices from other vendors. Refer to your device's installation guide for specifics or contact your Customer Support Center.

Common Applications

Assignable asynchronous lines can be used for the following purposes:

● To connect a PC, with PRIMELINK, to a Prime host when the host drives the PC.

● By NTS to initiate a connection request along a previously defined path between devices.

● For engineering applications such as terminal simulation and asynchronous line performance testing.

● When sensitive data or software is transmitted between systems. In this case, both systems assign the line number corresponding to their end of the line.

Security Considerations

As System Administrator, you may choose to enforce a Discretionary Access Control (DAC) policy with respect to assignable asynchronous lines. For example, you can prevent any user from assigning a line connected to a printer by setting a device ACL on the line itself.

For more information on developing a DAC policy for your site and instructions on setting device ACLs, see the *System Administrator's Guide, Volume III: System Access and Security.*

Assigning Asynchronous Lines

To assign an asynchronous line to a specific device, use the following procedure.

1. Include the NAMLC directive in your CONFIG file to reserve adequate buffers for the total number of assignable lines.

2. Use a SET_ASYNC -LINE *n* -ASSIGN YES command in your PRIMOS.COMI file to indicate that the line is assignable.

3. Include the AMLBUF directive in your CONFIG file to change the sizes of these buffers from their default settings (optional).

4. Issue an ASSIGN ASYNC -LINE *n* command from your terminal or the supervisor terminal to assign the line.

5. Issue a SET_ASYNC -LINE *n* [options] command from your terminal to change the speed, protocol, or configuration of the line (optional).

Note

Printers on the Spooler subsystem are assigned lines when the PROP -START command is issued. PROP -START only assigns lines for printers. You must assign lines for any other devices.

Chapter 4, Configuring Asynchronous Lines, provides a detailed explanation of the SET_ASYNC command. Chapter 3, Useful Configuration Directives, briefly describes the AMLBUF and NAMLC directives. For detailed information on the AMLBUF directive, read Chapter 6, Allocating Buffers. If you have questions concerning the PRIMOS.COMI file, consult the *System Administrator's Guide, Volume I: System Configuration.*

The ASSIGN ASYNC Command

The ASSIGN command gives you complete and uninterruptable control over a device. This prevents any other user, process, or device from using the assigned resource until you are finished. There are several valid device types, such as disks, tape drives and, in this case, asynchronous communication lines. The *PRIMOS Commands Reference Guide* contains a complete description of the ASSIGN command for all device types.

At Rev. 21.0, the decimal ASSIGN ASYNC command replaces the octal ASSIGN AMLC command. ASSIGN ASYNC allows you to assign a line or a range of consecutive lines in one command line. ASSIGN ASYNC applies the transparent protocol, TRAN, to all lines. If you want to change the protocol, speed, or configuration of the line, issue the appropriate SET_ASYNC command from your terminal. These changes remain in effect until you issue the UNASSIGN command or log out. Before you can assign an NTS line, it must be mapped by the System Administrator with the NTS_ASSOCIATE command. NTS users are allowed twenty seconds to complete each connection request before the ASSIGN command terminates.

The format of the ASSIGN ASYNC command is as follows:

ASSIGN -LINE n [-TO m]

Option	*Description*
-LINE n	The first line number in a series to be assigned where *n* specifies the required *decimal* line number to be assigned, or, when used with the -TO option. Remember lines 254 and 255 can only be used as assignable lines. All other lines can be either assignable or login lines.
-TO m	Conveniently assigns a consecutive range of lines from *n* to *m*, inclusive, where *n* and *m* are *decimal* line numbers.

For example, to assign line number 8 and change the protocol from TRAN, the default, to TTY, issue the following commands:

```
ASSIGN ASYNC -LINE 8
SET_ASYNC -LINE 8 -SPEED 9600 -PROTOCOL TTY
```

These two commands replace the old command format,

```
ASSIGN AMLC TTY 10 2413
```

To save time when you are assigning several consecutive lines, you can issue one ASSIGN ASYNC command as in the following example:

```
ASSIGN  ASYNC -LINE 250 -TO 255
```

Note

Although command lines issued in the old octal format are supported at this revision, the new decimal format is recommended.

The UNASSIGN ASYNC Command

UNASSIGN ASYNC, a new format for the UNASSIGN command, is available as a companion to the ASSIGN ASYNC command. Exclusive use of an asynchronous line is relinquished when you issue an UNASSIGN ASYNC command. Line numbers are specified in decimal. If you have assigned a range of consecutive lines, they can be released with one command line using the -TO option.

Use the UNASSIGN command to break a permanent connection over an NTS line. NTS users are allowed twenty seconds to complete each disconnect request before the UNASSIGN command terminates.

Users can unassign only those lines that they have assigned. The operator or System Administrator has the rights to unassign any line or device from the supervisor terminal, regardless of who assigned it or from which terminal it was assigned.

The format of the UNASSIGN ASYNC command is as follows:

UNASSIGN ASYNC -LINE n [-TO m]

Option	*Description*
-LINE n	Where *n* specifies the required *decimal* line number to be unassigned, or, when used with the -TO option, the first line number in a series to be released.
-TO m	Unassigns a consecutive range of lines from *n* to *m*, inclusive, where *n* and *m* are *decimal* line numbers.

For example, to release a range of five lines from line number 8 to line number 12, issue the following command:

```
UNASSIGN ASYNC -LINE 8 -TO 12
```

If you have assigned several lines, you may, of course, issue individual UNASSIGN ASYNC commands for each line when you are finished using it.

For example, lines 250 to 255 are assigned lines. Lines 250 and 253 are used for transmitting financial records from the host to branch offices. When the transmission is complete, these lines are released with the following command lines:

```
UNASSIGN ASYNC -LINE 250
UNASSIGN ASYNC -LINE 253
```

Lines 251, 252, 254, and 255 remain assigned to their processes or devices, and lines 250 and 253 are now available.

Note

Do not follow a decimal ASSIGN ASYNC command with an octal UNASSIGN AMLC. In limited situations this works, but it is generally better not to combine octal and decimal formats.

ASSIGN and UNASSIGN Error Messages

Before they allow an asynchronous line to be assigned or released, the ASSIGN ASYNC and UNASSIGN ASYNC commands verify your command line options. If an error, typically an omitted or invalid command line option, is detected, the line is not assigned. One of the error messages listed below is displayed, and you are returned to command level.

`Connection request rejected. (ASSIGN)`
> A request to assign a LTS line was refused by the LTS. This may be due to the fact that the line has a connection currently established, as a terminal connected into either this or another Prime system, or as an assigned line by another Prime system.

`Connection request timed out. (ASSIGN)`
> More than 20 seconds has expired during an attempt to assign an LTS line. This could indicate that the target LTS is either powered down or does not exist on the LAN.

`Device not assigned. (ASSIGN)`
> A user attempted to unassign a device that had not been assigned.

`Device output queue full. (ASSIGN)`
> The connection management output queue did not process the connect or disconnect request block generated by the ASSIGN or UNASSIGN command. This error message may indicate either a temporary failure of PRIMOS to communicate with the LHC controller, or a problem with the LHC controller itself. This may be a temporary problem. If the command is reissued, it may succeed.

`Improper command usage or arguments. (ASSIGN)`
> An incorrect option was specified in the command line or an argument was missing.

`Invalid range specified. (ASSIGN)`
> The lines, included in the range specified, are not assignable lines or one or both end points are out of range.

`Line not associated. (ASSIGN)`
> The ASSIGN command was issued to assign an NTS PRIMOS line number that has not been previously associated with an NTS_ASSOCIATE command.

Line number must be specified. (ASSIGN)

An ASSIGN ASYNC or UNASSIGN ASYNC command was issued without specifying a line number. The line number cannot be omitted from the command line.

Line number out of range. (ASSIGN)

The line number specified was less than 0 or greater than 256 for local lines or, for NTS lines, less than 1024 or greater than 1279.

No room. (ASSIGN)

The maximum number of assignable lines allocated by NTSASL or NAMLC is currently assigned. Either UNASSIGN a currently assigned line, or increase NTSASL and cold start the system.

No XCB available for request. (ASSIGN)

This message usually indicates a software failure, either in PRIMOS or the LHC. Contact your Customer Support Center.

NTS not started. (ASSIGN)

The ASSIGN or UNASSIGN command was issued for a NTS line before NTS was started.

Request queue full. (ASSIGN)

The connection management queue is full, and has not had an available entry for approximately 10 seconds. This usually indicates a NTS connection manager (NTS_SERVER) failure.

The device is in use. (ASSIGN)

The ASSIGN command was issued for a terminal line or an assignable line that is currently assigned by another user. This message is also seen when an UNASSIGN command is issued for a terminal line or a line that was assigned by another user.

The NTS_ASSOCIATE Command

The NTS_ASSOCIATE command defines a permanent or temporary path from a physical line attached to an LTS to a logical line number in the Prime host. As System Administrator, you must associate a line before a user can issue an ASSIGN, UNASSIGN, or interactive SET_ASYNC command for the line.

Note

The NTSDIM polls all NTS lines consecutively. Choose NTS assignable line numbers that are close to 1024 plus NTSUSR to avoid wasting system resources servicing unused lines and to improve performance.

The format of the NTS_ASSOCIATE command is

NTS_ASSOCIATE $\left\{ \begin{array}{l} \text{-LINE number -LTS\_NAME name -LTS\_LINE number} \\ \text{-PERMANENT} \\ \text{-HELP} \end{array} \right\}$

The NTS_ASSOCIATE command has the following arguments:

Argument	Description
-LINE number	Indicates the *decimal* line number in the Prime host. Valid NTS line numbers range from 1024 through 1279. Assignable NTS line numbers usually start at 1024 plus the decimal value of NTSUSR. Although any NTS line can be made assignable by issuing a SET_ASYNC -LINE n -ASSIGN yes command.
$\left\{\begin{array}{l}\textbf{-LTS\_NAME name}\\ \textbf{-LNAME}\end{array}\right\}$	Specifies the logical name of a particular LTS. The Network Administrator gives each LTS on a LAN a unique name when it is added to the configuration with CONFIG_NTS. LTS names can be a maximum of 16 characters long and must follow the same format and rules as PRIMENET node names. You can list all the LTS names on your LAN with the LIST_LTS_STATUS command.
$\left\{\begin{array}{l}\textbf{-LTS\_LINE number}\\ \textbf{-LLINE}\end{array}\right\}$	Specifies the LTS line number. *number* can range from 0 through 7.
$\left\{\begin{array}{l}\textbf{-PERMANENT}\\ \textbf{-PERM}\end{array}\right\}$	Establishes a stable connection that is reconnected automatically if a disconnect occurs. A temporary connection can not recover if the carrier signal is lost for three minutes and is unassigned automatically. A connection is permanent until the line is unassigned. A permanent connection can be removed by reissuing the NTS_ASSOCIATE command without supplying this option. Permanent connections are recommended for spooler lines.
$\left\{\begin{array}{l}\textbf{-HELP}\\ \textbf{-H}\end{array}\right\}$	Displays the command format and options.

For more information on the NTS_ASSOCIATE command, see the *NTS User's Guide.* For more information on CONFIG_NTS, see the *NTS Planning and Configuration Guide.*

Examples of the NTS_ASSOCIATE Command

Here are some examples of the NTS_ASSOCIATE command.

Making a Temporary Association: The following NTS_ASSOCIATE command issued from the supervisor terminal maps the logical PRIMOS line number 1153 to the physical line number 1 attached to the LTS SCIENCE.LAB.

```
OK, NTS_ASSOCIATE -LNAME SCIENCE.LAB -LLINE 1 -LINE 1153
[NTS_ASSOCIATE Rev. 21.0 Copyright (c) 1986, Prime Computer, Inc.]
OK,
```

Making a Permanent Association: The NTS_ASSOCIATE command can create a permanent association to an LTS line supporting a printer. In the following example, the association between the PRIMOS line number 1154 and the printer attached to line 0 of the LTS MATH.DEPT is stable. The link to the printer recovers from network failures automatically.

```
OK, NTS_ASSOCIATE -LNAME MATH.DEPT -LLINE 0 -LINE 1154 -PERM
[NTS_ASSOCIATE Rev. 21.0 Copyright (c) 1986, Prime Computer, Inc.]
OK,
```

It is also possible to make an existing association permanent. Reissue the NTS_ASSOCIATE command with the -PERM option and specify the PRIMOS line number as shown in the next example.

```
OK, NTS_ASSOCIATE -LNAME MATH.DEPT -LLINE 1 -LINE 1155
[NTS_ASSOCIATE Rev. 21.0 Copyright (c) 1986, Prime Computer, Inc.]
OK, NTS_ASSOCIATE -LINE 1155 -PERM
[NTS_ASSOCIATE Rev. 21.0 Copyright (c) 1986, Prime Computer, Inc.]
OK,
```

The NTS_LIST_ASSOCIATE Command

The NTS_LIST_ASSOCIATE command displays the current associations between PRIMOS line numbers and LTS line numbers. Active connections can be specified by PRIMOS line number or by LTS name and LTS line number. If the command is issued without options, information for all associations is displayed. Permanent connections are indicated in the display by the letter P.

The format of the NTS_LIST_ASSOCIATE command is

NTS_LIST_ASSOCIATE $\left\{ \begin{array}{l} \text{-LINE number} \\ \text{-LTS\_NAME name [-LTS\_LINE number]} \\ \text{-HELP} \end{array} \right\}$

The NTS_LIST_ASSOCIATE command has the following arguments:

Argument	Description
-LINE number	Indicates the *decimal* line number in the Prime host. Valid NTS line numbers range from 1024 through 1279.
{ -LTS_NAME name { -LNAME	Specifies the logical name of a particular LTS. You can list all the LTS names on your LAN with the LIST_LTS_STATUS command.
{ -LTS_LINE number { -LLINE	Specifies the LTS line number. *number* can range from 0 through 7.
{ -HELP { -H	Displays the command format and options.

For more information on the NTS_LIST_ASSOCIATE command, see the *NTS User's Guide*.

Examples of the NTS_LIST_ASSOCIATE Command

Here are some examples of the NTS_LIST_ASSOCIATE command.

Listing All Active Associations:

```
OK, NTS_LIST_ASSOCIATE
[NTS_LIST_ASSOCIATE Rev. 21.0 Copyright (c) 1986, Prime Computer, Inc.]

Primos                   LTS
line #      LTS Name     line #
 1153       SCIENCE.LAB    1
 1154  [p]  MATH.DEPT      0
 1155  [p]  MATH.DEPT      1

* [p] denotes a permanent association.
OK,
```

Listing Associations for an LTS:

```
OK,NTS_LIST_ASSOCIATE -LNAME MATH.DEPT
[NTS_LIST_ASSOCIATE Rev. 21.0 Copyright (c) 1986, Prime Computer, Inc.]

Primos                   LTS
line #      LTS Name     line #
 1154  [p]  MATH.DEPT      0
 1155  [p]  MATH.DEPT      1

* [p] denotes a permanent association.
OK,
```

Listing Associations for an Individual Line:

```
OK, NTS_LIST_ASSOCIATE -LINE 1153
[NTS_LIST_ASSOCIATE Rev. 21.0 Copyright (c) 1986, Prime Computer, Inc.]

Primos                       LTS
line #     LTS Name          line #
 1153      SCIENCE.LAB          1
OK,
```

The NTS_UNASSOCIATE Command

The NTS_UNASSOCIATE command removes a permanent or temporary path from a physical line attached to an LTS to a logical line number in the Prime host. Only the System Administrator can unassociate a line. If a line is currently assigned by another user, it can not be unassociated until it is unassigned.

There are two ways to remove a connection: by specifying the *primos_line_number* or by specifying the *lts_name* and *lts_line_number*. Use the NTS_LIST_ASSOCIATE command to display a list of all currently associated lines.

The format of the NTS_UNASSOCIATE command is

$$\text{NTS\_UNASSOCIATE} \left\{ \begin{array}{l} \text{-LINE number} \\ \text{-LTS\_NAME name} \quad \text{-LTS\_LINE number} \\ \text{-HELP} \end{array} \right\}$$

The NTS_UNASSOCIATE command has the following arguments:

Argument	*Description*
-LINE number	Indicates the *decimal* line number in the Prime host. Valid NTS line numbers range from 1024 through 1279.
-LTS_NAME name **-LNAME**	Specifies the logical name of a particular LTS. You can list all the LTS names on your LAN with the LIST_LTS_STATUS command.
-LTS_LINE number **-LLINE**	Specifies the LTS line number. *number* can range from 0 through 7.
-HELP **-H**	Displays the command format and options.

For more information on the NTS_UNASSOCIATE command, see the *NTS User's Guide.*

Examples of the NTS_UNASSOCIATE Command

Here are some examples of the NTS_UNASSOCIATE command.

Unassociating an LTS line: The following command line breaks the connection between the host and line 1 on the LTS MATH.DEPT.

```
OK, NTS_UNASSOCIATE -LNAME MATH.DEPT -LLINE 1
[NTS_UNASSOCIATE Rev. 21.0 Copyright (c) 1986, Prime Computer, Inc.]
OK,
```

Unassociating a PRIMOS line: The following command line releases the PRIMOS line number 1153 from any active association.

```
OK, NTS_UNASSOCIATE -LINE 1153
[NTS_UNASSOCIATE Rev. 21.0 Copyright (c) 1986, Prime Computer, Inc.]
OK,
```

NTS_ASSOCIATE, NTS_LIST_ASSOCIATE, and NTS_UNASSOCIATE Error Messages

If the syntax of the command line is incorrect, a resource is not available, or there is a problem with NTS, one of the error messages below is displayed and the command is rejected.

Both an LTS name and an LTS line number are required. (NTS_UNASSOCIATE)
> One of these options was issued without the other on the command line. No action is taken on this command.

Either a PRIMOS line number, or an LTS name and line number is required.(NTS_UNASSOCIATE)
> No command arguments were issued on the NTS_UNASSOCIATE command line.

Insufficient access rights. (NTS_ASSOCIATE)
> An attempt has been made by an unauthorized user to associate a NTS PRIMOS line.

Insufficient access rights. (NTS_UNASSOCIATE)
> An attempt has been made by an unauthorized user to unassociate a NTS PRIMOS line.

LTS line already associated. (NTS_ASSOCIATE)
> An attempt has been made to associate an LTS that is already associated with another PRIMOS line number.

LTS line not associated. (NTS_UNASSOCIATE)
> An attempt has been made to unassociate an LTS that is not associated.

-LTS_LINE argument is not permitted without -LTS_NAME (NTS_LIST_ASSOCIATE)
> Command was issue with insufficient arguments to identify the line's LTS.

Line <n> is assigned to username <user ddd>; it must be unassigned first.
(NTS_UNASSOCIATE)
> The System Administrator has made an attempt to unassociate PRIMOS line number n that is already assigned to username (user *ddd* (decimal)). It must be unassigned first. Any line may be unassigned from the supervisor terminal using the UNASSIGN ASYNC command.

`Line is not associated. (NTS_LIST_ASSOCIATE)`
> Command was issued for an unassociated PRIMOS line number.

`Line not associated. PRIMOS line number, -LTS_NAME, and -LTS_LINE arguments must be`
`supplied. (NTS_ASSOCIATE)`
> An attempt has been made to associate a previously unassociated PRIMOS line number without all the proper arguments. In this case, all three arguments must be supplied.

`No associations (NTS_LIST_ASSOCIATE)`
> There are no active associations on NTS.

`No associations with the specified LTS (NTS_LIST_ASSOCIATE)`
> There are no active associations for the specified LTS.

`No associations with the specified LTS and LTS line number. (NTS_LIST_ASSOCIATE)`
> There is no active association for the line number indicated on the specified LTS.

`Network Terminal Service is not started. (NTS_ASSOCIATE)`
> An attempt has been made to make an association when network terminal service is not currently started.

`Network Terminal Service is not started. (NTS_LIST_ASSOCIATE)`
> The command was issued when network terminal service was not currently started.

`Network Terminal Service is not started. (NTS_UNASSOCIATE)`
> An attempt has been made to break an association when network terminal service is not currently started (or not configured).

`No other options may be specified with -HELP. (NTS_ASSOCIATE)`
> The command issued when -HELP is found on the command line with -LINE, -LTS_NAME, or -LTS_LINE.

`No other options may be specified with -HELP. (NTS_LIST_ASSOCIATE)`
> The command issued when -HELP is found on the command line with -LINE, -LTS_NAME, or -LTS_LINE.

`No other options may be specified with -HELP. (NTS_UNASSOCIATE)`
> The command issued when -HELP is found on the command line with -LINE, -LTS_NAME, or -LTS_LINE.

`No NTS assignable lines are configured. (NTS_ASSOCIATE)`
> An attempt has been made to associate a NTS PRIMOS line when there are no NTS assignable lines configured in the system (NTSASL = 0). This configuration directive must be changed if NTS assignable lines are to be utilized, and the system cold started.

`PRIMOS line already associated. (NTS_ASSOCIATE)`
> An attempt has been made to associate a PRIMOS line number that is already associated.

`PRIMOS line number is required. (NTS_ASSOCIATE)`
> A PRIMOS line number has not been specified in the command when required.

PRIMOS line not associated. (NTS_UNASSOCIATE)
> An attempt has been made to unassociate a PRIMOS line number that is not associated.

PRIMOS line number not permitted with -LTS_NAME. (NTS_LIST_ ASSOCIATE)
> The PRIMOS line number must be specified alone.

The LTS line number must be between 0 and 7. (NTS_ASSOCIATE)
> An attempt has been made to issue an NTS_ASSOCIATE command for an LTS line number that is out of range. Valid LTS line numbers currently range from 0 through 7.

The LTS line number must be between 0 and 7. (NTS_LIST_ASSOCIATE)
> An attempt has been made to issue an NTS_LIST_ASSOCIATE command for an LTS line number that is out of range. Valid LTS line numbers currently range from 0 through 7.

The LTS line number must be between 0 and 7. (NTS_UNASSOCIATE)
> An attempt has been made to issue an NTS_UNASSOCIATE command for an LTS line number that is out of range. Valid LTS line numbers are currently from 0 through 7.

The -LTS_NAME and -LTS_LINE arguments are not allowed when a PRIMOS line number is specified. (NTS_UNASSOCIATE)
> Issued when an NTS_UNASSOCIATE command contains a PRIMOS line number and either an LTS name or an LTS line number. Reissue the command with just the PRIMOS line number, or both the LTS name and line number.

The PRIMOS line number must be between 1024 and 1279. (NTS_ASSOCIATE)
> An attempt has been made to associate a PRIMOS line that is not in the NTS line range. These lines cannot be associated.

The PRIMOS line number must be between 1024 and 1279. (NTS_LIST_ASSOCIATE)
> The command was issued for a PRIMOS line that is not in the NTS line range.

The PRIMOS line number must be between 1024 and 1279. (NTS_UNASSOCIATE)
> An attempt has been made to unassociate a PRIMOS line that is not in the NTS line range. These lines cannot be associated or unassociated.

The specified LTS is not configured. (NTS_ASSOCIATE)
> An attempt has been made to use an LTS name that has not been previously configured with a MAC address (by CONFIG_NTS). If necessary, this LTS can be added with NTS started by CONFIG_NTS (without stopping NTS).

The specified LTS is not configured. (NTS_LIST_ASSOCIATE)
> An attempt has been made to use an LTS name that has not been previously configured with a MAC address (by CONFIG_NTS). If necessary, this LTS can be added with NTS started by CONFIG_NTS (without stopping NTS).

The specified LTS is not configured. (NTS_UNASSOCIATE)
> An attempt has been made to use an LTS name that has not been previously configured with a MAC address (by CONFIG_NTS). If necessary, this LTS can be added with NTS started by CONFIG_NTS (without stopping NTS).

The specified PRIMOS line is not assignable. (NTS_ASSOCIATE)
>   An attempt has been made to associate a PRIMOS line that is not currently assignable.

When -LTS_NAME or -LTS_LINE is specified, the other is required. (NTS_ASSOCIATE)
>   An attempt has been made to issue an NTS_ASSOCIATE command without both LTS name and LTS line number. This form of the command requires the presence of both.

6

ALLOCATING I/O BUFFERS

The PRIMOS operating system allocates storage locations in memory, called I/O buffers, for every interactive user process. I/O buffers effectively compensate for differences between device baud rates and internal event timers. The first part of this chapter discusses how PRIMOS uses I/O buffers to store, forward, and manage the flow of characters.

I/O buffers are reserved for users in the following order:

1. Local terminal users

2. PRIMENET remote login users

3. NTS terminal users

4. Local assignable lines

5. NTS assignable lines

The default buffer sizes are large enough for most devices to operate efficiently. However, as System Administrator, you can fine tune your system for any device or application that requires more or less space. The second part of this chapter explains how to use the AMLBUF directive to change the size of buffers for local terminal and assignable lines.

The REMBUF, NTSBUF, and NTSABF directives define the size of I/O buffers for PRIMENET remote login users, NTS terminal users, and NTS assignable lines, respectively. For more information on these directives see Chapter 3, Useful Configuration Directives.

Asynchronous Character Processing

When you type a character, it is transmitted from the keyboard over an asynchronous line to a communications controller board. Asynchronous devices, such as terminals, are connected to either AMLC or ICS controllers or an LTS, which communicates to the host through an LHC controller. Figure 6-1 illustrates the flow of data from the input device to PRIMOS.
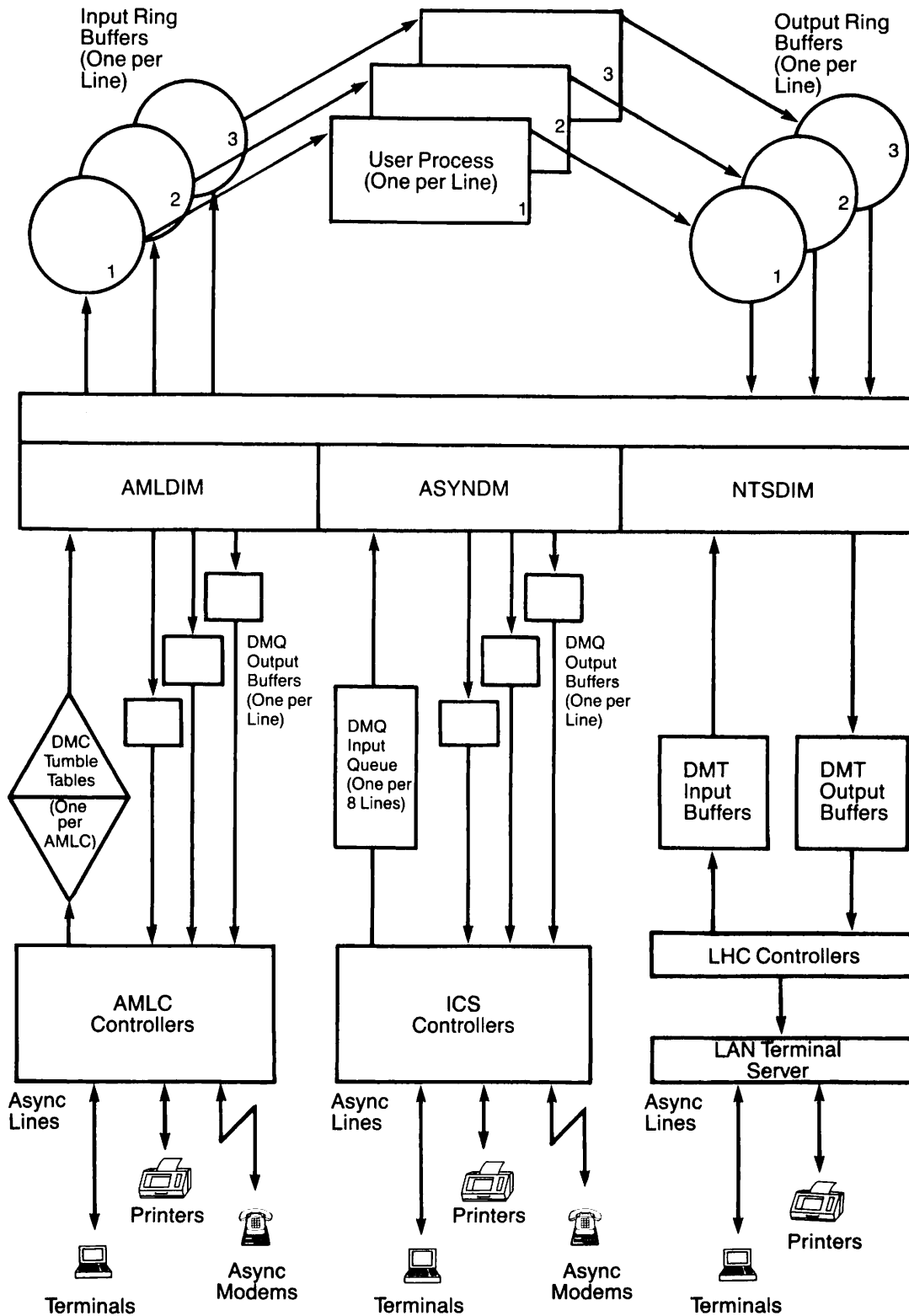
FIGURE 6-1. Input and Output Buffers and Queues for Asynchronous Lines

AMLC, ICS, and LHC controller boards receive characters from many lines and store them in either a Direct Memory Channel (DMC) tumble table, a Direct Memory Queue (DMQ) input queue, or a Direct Memory Transfer (DMT) input buffer, respectively.

Each type of controller is supported by a special process known as an Asynchronous Device Interface Module (ASYNC DIM). The three ASYNC DIMs are AMLDIM, ASYNDM, and NTSDIM. Periodic I/O interrupts trigger the ASYNC DIMs to read incoming data, analyze flow control signals, and perform any per-character processing. In general, the ASYNC DIMs decode the bits that designate line number and protocol before transferring the character to the appropriate area in main memory.

Each asynchronous line has a fixed area in memory called an Input Ring Buffer (IRB). IRBs have pointers to indicate positions within the buffer as follows: write pointers indicate to the ASYNC DIMs where to put the next character, read pointers indicate to PRIMOS which character to process next. IRB characters are processed and any output characters are passed to the appropriate Output Ring Buffer (ORB). Both types of ring buffer are referenced by an index number. For login lines, this index number is the same as the user number. For assignable lines, the index number refers to a pair of buffers from a rotating group of shared ring buffers.

The ASYNC DIMs retrieve the data from the ORB, identify the target line number, apply the correct communications protocol for the output device (for example TTYUPC), and pass the character to the appropriate DMQ or DMT output buffer. The communication controllers accept data from the output buffers at regular intervals and separate the characters by line numbers. The controllers then transmit each character to the specified device at the correct baud rate.

DMQ output buffers are maintained for each local line. DMT output buffers are maintained for each LHC controller. IRBs and ORBs are maintained for each user. These buffers reside permanently in memory (that is, they are wired), even when a particular line is not in use.

If you do not specify buffer sizes in the system configuration file, PRIMOS uses the default values. The default IRB and ORB values are sufficient for most devices, but the default DMQ input size can be too small when the line speed is greater than 1200 baud.

Block-mode terminals, asynchronous links from one computer to another, serial graphics devices, and certain printers require larger buffer sizes to operate efficiently. Refer to the installation guide for your device for the buffer sizes recommended by the manufacturer, or consult your Customer Support Center.

The AMLBUF directive defines the size of the IRBs, ORBs, and DMQ buffers for local terminal and assignable lines. Instructions for changing buffer sizes are provided in the section, The AMLBUF Directive, on page 6-7.

The REMBUF, NTSBUF, and NTSABF directives define the size of I/O buffers for PRIMENET remote login users, NTS terminal users, and NTS assignable lines, respectively.

The DMC tumble tables and DMQ input buffer sizes are set with the AMLIBL and ICS INPQSZ directives, respectively. For more information on these directives see Chapter 3, Useful Configuration Directives.

Default Buffer Number Assignments

The DMQ ring buffer number for login lines is permanently associated with the physical line number. The default buffer number for any login line is equal to the physical line number plus two. Although it is possible to change the default buffer assignment with a SET_ASYNC -LINE *n* -USER_NUMBER *m* command, this is not recommended.

I/O Interrupt Rates

I/O interrupt rates determine how often the controllers interrupt the ASYNC DIMs to service the I/O buffers and queues shown in Figure 6-1. The default interrupt rate for AMLC and ICS controllers is ten interrupts per second. Ideally, this rate is sufficient to service the buffers without interrupting the ASYNC DIMs too frequently. A higher interrupt rate increases the amount of CPU time spent servicing the buffers and thus reduces performance.

In some cases performance can be improved by changing the size of the I/O buffers and queues. The sections on determining buffer capacity explain how to do this. The default interrupt cycle can cause perceptible delays in a highly interactive graphics package. Increasing the interrupt rate to 30 per second improves response.

On a system with both AMLC and ICS controllers, the baud rate of the last physical AMLC line controls the rate for processing AMLC interrupts. The ICS INTRPT directive sets the rate for ICS interrupts.

The AMLC I/O Interrupt Rate

Each AMLC controller reads incoming characters from all the lines attached to it and transfers them to a two-part storage area called a DMC tumble table. See Figure 6-1. When the current tumble table buffer is full, the AMLC controller switches the incoming stream of characters to the other tumble table buffer and interrupts the AMLDIM process. The controllers also interrupt the AMLDIM on a regular basis. The frequency of these interrupts is determined by the baud rate of the last physical AMLC line. When the AMLDIM process executes, it reads each character in the tumble table, identifies the line that it came from, and transfers it to the line's input ring buffer. If the line is full-duplex, the characters are also sent to the line's output ring buffer.

Caution

The default AMLC I/O interrupt rate is 10 interrupts per second. This requires that the last AMLC line be configured for 110 baud. Increasing this line's baud rate beyond 300 baud (30 interrupts per second) results in a *severe* performance degradation. For this reason, do not assign this line or connect it to a high speed device. Reserve this line for a very slow terminal or a printer. If your hardware resources permit, configure this line at 110 baud but do not connect it.

The AMLC I/O interrupt rate is calculated by dividing the baud rate of the system's highest-numbered AMLC line (that is, the last line of the last AMLC board) by the character size (11 bits for 110 baud, 10 bits for other speeds).

The ICS I/O Interrupt Rate

The I/O interrupt rate for ICS controllers is set by the ICS INTRPT directive. It is independent of the speed or use of any ICS lines. The default and minimum value is 12_8 (10) interrupts per second. The maximum is 144_8 (100) interrupts per second. If the value selected is out of range (too high or too low), the ICS controller defaults to the maximum or minimum value, as appropriate. The procedure for calculating the ICS interrupt rate is explained in Chapter 3, Useful Configuration Directives, and in the *ICS User's Guide.*

Determining IRB Capacity

OAS, FORMS, PRIMEWAY™, DPTX, and PRIME/SNA™ products use block-mode terminals, which require large IRBs. The default IRB size allows 255 type-ahead characters on terminals using character-mode input. When a burst-type device sends more than 256 characters in a burst, the buffer overflows and characters are lost. The overflow characters are not echoed back to the screen. If the terminal is in block mode when the IRB overflows and the screen termination characters are discarded, the terminal may lock.

The recommended IRB sizes for these products are as follows:

Application	Input Buffer Size
OAS on a PT65	1500_8 (832)
DPTX	3700_8 (1984)
FORMS/PRIMEWAY	1500_8 (832) or the maximum screen size from the application FD plus twice the number of fields on screen
PRIME/SNA	2000_8 (1024) (24 x 80 only) 4000_8 (2048) (24 x 80) and (27 x 132)

Determining ORB Capacity

The default ORB size is generally sufficient for all character-mode terminals. If full screens of data are painted at 9600 baud or greater, the ORB can be made larger to increase throughput. Setting the ORB size too small can reduce throughput for high-speed devices. Perceptible bursts of output can occur on block-mode terminals using the following applications: OAS on the PT65, FORMS, PRIMEWAY, DPTX, and PRIME/SNA. PRIME MEDUSA, EDMS™, and other products that send serial graphic output over asynchronous lines require an ORB size greater than the maximum screen size; 3720_8 (2000) is recommended.

For improved throughput, set the ORB capacity to equal or exceed the number of characters that the line can transmit in 1/2 second.

ORB capacity in halfwords = $\dfrac{\text{characters per second}}{2}$

For example, a 9600-baud line sends 960 10-bit characters per second to a laser printer. The optimal ORB size for this line is 360_8 (240) halfwords.

Determining DMQ Buffer Capacity

The DMQ buffer capacity depends on the line speed, character length, and the interrupt rate. The DMQ buffer stores 1 character per halfword. Legal values are 2000_8, 1000_8, 400_8, 200_8, 100_8, 40_8, and 20_8. Use the following formula to calculate the DMQ size and use the next higher value listed in Table 6-1.

DMQ Size = $\dfrac{\text{line speed divided by bits per character}}{\text{I/O interrupt rate per second}}$

For example,

DMQ Size = $\dfrac{9600 \text{ divided by } 10}{10}$ = 96

Convert the result of 96 to octal, 140_8, and round up to the next valid size, which is 200_8 (128).

When DMQ buffer capacity is too small, the effective speed of the device decreases. When the DMQ capacity is too large, devices with small input buffers can overflow. Serial printers often either have very small internal buffers, or wait until their buffers are nearly full before sending an XOFF (or using DSS) to tell PRIMOS to stop sending output.

Table 6-1 shows the recommended DMQ buffer sizes when the default interrupt rate (10) is used.

TABLE 6-1.   DMQ Buffer Capacity

Line Speed (Baud rate)	Characters Per Second	Recommended DMQ Capacity
50	5	20_8 (16)
75	7	20_8 (16)
110	10	20_8 (16)
134.5	13	20_8 (16)
150	15	20_8 (16)
200	20	20_8 (16)
300	30	20_8 (16)
600	60	20_8 (16)
1200	120	20_8 (16)
1800	180	40_8 (32)
2400	240	40_8 (32)
3600	360	100_8 (64)
4800	480	100_8 (64)
7200	720	200_8 (128)
9600	960	200_8 (128)
19200	1920	400_8 (256)

The AMLBUF Directive

The AMLBUF directive defines the capacities of the Input Ring Buffers (IRB), Output Ring Buffers (ORB), and Direct Memory Queue (DMQ) output buffers for login and assignable lines. AMLBUF directives are stored in your PRIMOS configuration file and set buffer sizes at cold start. All arguments to the AMLBUF directive are octal numbers.

AMLBUF directives are necessary only for lines connected to devices that do not run efficiently with the default buffer sizes. This directive has two distinct formats as shown below. Each login line requires an individual AMLBUF directive. Assignable lines may require two directives.

For login lines, only one AMLBUF directive is necessary:

    AMLBUF   line-number   IRB-size   ORB-size   DMQ-size

Assignable lines may require two AMLBUF directives. The first AMLBUF directive sets only the DMQ size and the second sets the ring buffer sizes.

    AMLBUF   line-number   0   0   DMQ-size

    AMLBUF   index   IRB-size   ORB-size   0

Caution

The system dynamically allocates ring buffers for assigned lines. There is no guarantee that the same buffers are used when a line is unassigned and later reassigned. It is only possible at cold start to determine which ring buffers a particular line receives. Therefore, it is recommended that you use the AMLBUF directive to set the ring buffers for *all* assignable lines to the largest size that any individual device requires.

If any of the size arguments to the AMLBUF directive are set to zero or omitted, the default size or, when there is an existing AMLBUF directive, the last value specified is used.

The AMLBUF directive has the following arguments:

Argument	*Description*
line-number	The physical line number; required for setting the DMQ buffer.
index	The value used by PRIMOS to reference ring buffers for assignable lines. Index ranges from (NTUSR + NRUSR -1) to (NTUSR + NRUSR -1 + NAMLC).
IRB-size	The size in halfwords (two characters per halfword) of the input ring buffer. The minimum value is 1 and the maximum is 7777_8 (4095). The default value is 200_8 (128).
ORB-size	The size in halfwords (two characters per halfword) of the output ring buffer. The minimum value is 62_8 (50) and the maximum is 7777_8 (4095). The default value is 300_8 (192).
DMQ-size	The size in halfwords (one character per halfword) of the DMQ output buffer. The value must be a power of 2 from 20_8 (16) through 2000_8 (1024), inclusive. If you specify a value less than 20_8 (16), the system sets the size to 20_8 (16). The default value is 40_8 (32). The sum of all DMQ buffers on the system cannot exceed 64,000 halfwords.

You can add comments at the end of each AMLBUF directive to make your CONFIG file easier to read and modify.

For another description of AMLBUF, See Chapter 3, Useful Configuration Directives.

The AMLBUF directives are included in your CONFIG file. The sample AMLBUF directives shown in Figure 6-2 set the IRB, ORB, and DMQ buffer sizes for a system with 20 asynchronous lines.

```
/*    Lines 0 to 3 are using the default buffer sizes.
/*    Buffer assignments login lines 4, 5, 6, 7, and 8.
/*    line number  IRB-size ORB-size DMQ-size
AMLBUF    4         1        62       20    /* Unused login line
AMLBUF    5         200      2000     40    /* New Medusa PW 153
AMLBUF    6         1500     500      200   /* PT65 running  OAS
AMLBUF    7         1000     1000     200   /* 9600 baud with DPTX
AMLBUF    10        1000     1000     200   /* 1200 baud with DPTX
/*
/*    DMQ buffer assignments for assignable lines
/*    line number  IRB-size ORB-size DMQ-size
AMLBUF    21        0        0        20    /* LQ Printer
/*
/*    Ring buffer assignments for assignable lines
/*    Set all IRBs and ORBs to the maximum size
/*    required by any individual device. 1024 ('2000)
/*         index   IRB-size ORB-size DMQ-size
AMLBUF    17        2000     2000     0     /* MEDUSA  PW 95, or
AMLBUF    20        2000     2000     0     /* computer link, or
AMLBUF    21        2000     2000     0     /* serial printer.
```

FIGURE 6-2.   Sample AMLBUF Directives

Figure 6-2 shows 9 login lines, 6 lines for PRIMENET remote login users, 3 assignable lines, and a supervisor terminal. Therefore, NTUSR = 12_8 (10), NRUSR = 6_8 (6), and NAMLC = 3_8 (3).

Login line number 4 is unused. The IRB, ORB, and DMQ buffers are set to their minimum values to conserve memory; 1_8, 62_8, 20_8 halfwords, respectively.

Login line 5 is connected to a PRIME MEDUSA workstation model number PW 153, which does require an assigned line. The default IRB and DMQ sizes are used and the ORB size is increased to 2000_8 (1024) halfwords.

Login line number 6 is connected to a PT65 terminal running OAS. The IRB and ORB sizes are increased to 1500_8 (832) halfwords and 500_8 (320) halfwords, respectively. The DMQ size is increased to 200_8 (128).

Login lines 7 and 10_8 (8) are configured for 9600 baud and 1200 baud, respectively. The DMQ buffer sizes for both lines are increased to 200_8 (128). These lines are used primarily for DPTX. Increasing the IRB and ORB values to 1000_8 (512) also increases throughput.

The remaining login lines are connected to devices that operate well with the default buffer sizes.

The three assignable lines are used for a bidirectional letter quality printer, a serial graphics line for PRIME MEDUSA PW 95 workstation, and an incoming only computer-to-computer link.

A bidirectional printer is connected to physical line number 21_8 (17). To prevent its small internal buffer from overflowing, use an AMLBUF directive to set the DMQ buffer to the minimum value, 20_8 (16).

Although the printer can use the default ring buffer sizes, the serial graphics line requires a large ORB and the computer-to-computer link requires a large IRB.

Using the second format given on page 6-7, the ring buffer *index* for assignable lines ranges from 17_8 (15) through 21_8 (17). Use three AMLBUF directives to set each pair of IRBs and ORBs to 2000_8 (1024) for all possible values of *index*: 17_8 (15), 20_8 (16), and 21_8 (17).

APPENDICES

A

OBSOLETE AND RARELY USED COMMANDS

This appendix describes the AMLC, ASSIGN AMLC, and UNASSIGN AMLC commands. These commands use octal bitstrings to configure, assign, and release asynchronous lines on AMLC, ICS, and LTS controllers.

Obsolete Commands

▶ AMLC

The AMLC command uses octal bitstrings to configure both terminal and assignable asynchronous lines. At Rev. 20.2, the AMLC command was replaced by the SET_ASYNC command, a more straightforward way of configuring your asynchronous lines. Although the AMLC command is still supported, its use is no longer recommended. The command defines the following:

- The protocol for the line
- The use of Auto Speed Detect (ASD) by the line
- The line speed, bit pattern, and parity for the line
- The initial terminal characteristics of the line
- The index number of buffers associated with the line

Caution

All numbers used in the AMLC command must be *OCTAL* numbers.

The format of the AMLC command is as follows:

<u>AM</u>LC [protocol] line [config_word [lword]]

Argument	Description
protocol	Defines the format and relative timing of data exchanged over the line, where *protocol* can be one of the following: ASD, TRAN, TTY, TT8BIT, TTYUPC, TTYNOP, TTY8, or TTY8HS. Do not use TTYNOP and ASD for NTS lines. Model number 5054 AMLC controller boards use TTYHS, TRANHS, or TTYHUP. Detailed definitions of each protocol are provided in the following section, Standard Protocols.
line	Specifies the octal line number. The maximum value is 377_8 (255) for lines attached to AMLC and ICS controllers. Line numbers 376_8 and 377_8 can not be used as login lines and must be configured as assignable lines. Valid NTS line numbers range from 2000_8 (1024) through 2377_8 (1279). Display line numbers with the STATUS USERS command described in Chapter 4, Configuring Asynchronous Lines.
config_word	An octal bit string that defines the configuration parameters, such as data set control, baud rate, reverse flow control, and parity. The bit pattern definitions and a template can be found in the section, Constructing Octal Bitstrings, on page A-5. The four most commonly used *config_words* and the baud rates associated with them are as follows:

config_word	*Baud Rate*
2033	110 Baud (uses 2 stop bits)
2213	300 Baud
2313	1200 Baud (default)
2413	Programmable clock value (clock default is 9600 Baud)

Argument	Description
lword	An octal bitstring that specifies full-duplex or half-duplex mode, recognizes XON and XOFF, and enables data set sense. For login lines, the *lword* associates a pair of buffers with the user number, an index to the location of the buffers.

Note

When the AMLC command is used to configure NTS lines, certain restrictions apply. These constraints are highlighted in the section, NTS Considerations, later in this appendix.

Standard Protocols

The values for the *protocol* argument are TRAN, TTY (the default), TT8BIT, TTYUPC, TTYNOP, TTY8, TTY8HS, TRANHS, and ASD. The basis for selection of the protocol is discussed below. Protocols for older model AMLC boards (model 5054) are discussed separately.

TRAN: TRAN, the transparent protocol, is usually used by lines connected to peripheral devices or to other computers. With transparent protocol, input is not echoed, the input of a line feed or carriage return does not provoke a response, and carriage returns are not transformed to line feeds. CONTROL-P does not have a special meaning under this protocol and is passed through to the program. All characters pass as data unless XON/XOFF flow control is set in bit three of the lword.

TTY: TTY, the terminal protocol, is the protocol assigned at cold start to lines controlling interactive terminals. With the TTY protocol, all input from the terminal is echoed if the line is set for full-duplex; a carriage return and a line feed are echoed following a carriage return. The high order bit (the *ASCII* parity bit) of each character input from the terminal is forced on. CONTROL-P and BREAK are interpreted as a QUIT command if the terminal is connected to the system as a user terminal.

If the terminal is connected to the PRIMOS operating system with an assignable line, CONTROL-P, BREAK, and line feed input from the terminal are ignored and discarded. A carriage return input from the terminal is transmitted as a new line (or line feed) to the program requesting input.

If you do not specify a protocol when using the AMLC command, TTY is assigned by the operating system.

TT8BIT: TT8BIT behaves in the same manner as the TTY protocol except that the high order bit (ASCII parity bit) is not forced on for each character input from the terminal. All control characters are handled in the same manner as the TTY protocol.

Use this protocol only if you have an Arabic DM5E/PLUS terminal.

To use this protocol, you must set bit 13 of the line *config_word* to 0 to enable parity, and also set bit 14 to 0 to select odd parity, as shown in the example below:

```
AMLC TT8BIT 5 2403  /* Line 5: odd parity enabled, 9600 baud
```

TTYUPC: TTYUPC, the uppercase translating protocol, is used to avoid sending lowercase output to terminals or peripheral devices that cannot print lowercase characters. This is the only difference between TTY and TTYUPC protocols.

TTYNOP: TTYNOP configures the asynchronous DIM to ignore all traffic on the line. If you have a noisy line to which no device is connected, use this protocol to ensure that the CPU does not waste time trying to interpret any noise coming from this line as commands that it must process. Do not specify TTYNOP for an NTS line.

TTY8 and TTY8HS: TTY8 adapts the standard terminal protocol TTY for devices that use Prime ECS. If you have model 5052 or 5054 AMLC controller boards, use TTY8HS, a functional equivalent of TTY8 specifically designed for this hardware.

ASD: ASD (Auto Speed Detect) allows PRIMOS to detect automatically the transmission speed of a user terminal. Depending on how you configure the line, PRIMOS can detect baud rates of 110, 300, 600, 1200, 2400, 4800, 9600, and 19200 bits per second. Do not specify ASD for an NTS line. If you enable ASD you must explicitly set the line's baud rate to 9600.

Note

When the system enables ASD, it reconfigures the line for character length 7, odd parity, and 1 stop bit, to prepare for the next user. If it is necessary, use the SET_ASYNC command interactively to change these line characteristics after you establish the connection. Interactive changes to line characteristics are in effect for the current session only; when you log out, ASD is reenabled.

Protocols for Older Model AMLC Boards

If you have lines attached to an older model 5054 AMLC board (also known as the DMT AMLC board), use TTYHS, TRANHS, or TTYHUP as the value for the AMLC command's *protocol* argument. TTYHS, TRANHS, and TTYHUP are commonly referred to as high-speed protocols. When you specify TTYHS, TRANSHS, or TTYHUP for an NTS line, the high speed protocol automatically is converted to its equivalent standard protocol: TTY, TRAN, or TTYUP.

Do not use these protocols on the following lines:

- Lines attached to model 5154 AMLC boards (also known as QAMLC or DMQ AMLC boards) or to ICS controllers

- Lines that normally have their character-time-interrupt flag always set (for example, the last line on the last AMLC)

Depending on the baud rate and the number of lines in the group, lines using high-speed protocols can greatly increase the system overhead.

TTYHS and TRANHS: TTYHS and TRANHS are used by lines connected to peripheral devices that can run at greater than 1200 baud, which is the standard terminal speed. For example, lines using the high-speed protocols can run at 9600 baud. TTYHS and TRANHS are the same as the protocols TTY and TRAN, with one exception; these protocols have a burst-mode effect on the output device. When the output buffer contains more than 40 characters, the line's character time interrupt flag is set. This flag remains set until the output buffer contains fewer than 40 characters.

TTYHUP: TTYHUP, the high-speed translating protocol, is used to avoid sending lowercase output to terminals or peripheral devices that cannot print lowercase characters.

Constructing Octal Bitstrings

The *config_word* and *lword* arguments to the AMLC command are octal numbers that correspond to 16-bit halfwords. The procedure for constructing an octal bitstring is as follows:

1. Decide how you want to configure the line, the speed, the protocol, and so forth.

2. Select the appropriate values from the list for each parameter and fill in the blanks for each bit in the *config_word* template (Figure A-1).

3. Write the octal value for each block of zeros and ones in the template in the spaces provided. A conversion table is supplied in Figure A-1 for your convenience.

4. Read the bit pattern from left to right. Leading zeros can be ignored.

5. Repeat steps 1 to 4 for the *lword* template.

Using Data Set Sense

Bits 5 and 6 of the *lword* are used to support devices that toggle an RS-232-C pin to signal when they are busy/ready, rather than using XON/XOFF. RS-232-C pin 8, Data Carrier Detect (DCD) is used for the signal.

Bit 5 of the *lword* indicates that pin 8 should be interrogated before performing an output operation. If pin 8 busy is detected, PRIMOS responds as if an XOFF was received for that line. When pin 8 goes to the ready state, it is flagged as if an XON was received, and output resumes. A device can signal busy by causing pin 8 to be raised or lowered.

Bit 6 is checked only if bit 5 is set. Bit 6 can be set to interrogate pin 8 either way. For example, if the device signals busy as pin 8 high (1), the *lword* bit setting is the following:

Bit 5 = 1 (Use data set sense.)

Bit 6 = 1 (If pin 8 is low, interpret as XOFF.
          If pin 8 is high, interpret as XON.)

On some devices, pins other than pin 8 may be used. If this is the case, arrange the cabling so that the data set sense signal is wired into carrier detect for the controller.

Data set sense is also referred to as buffered protocol or reverse channel protocol.

Building the Configuration Word
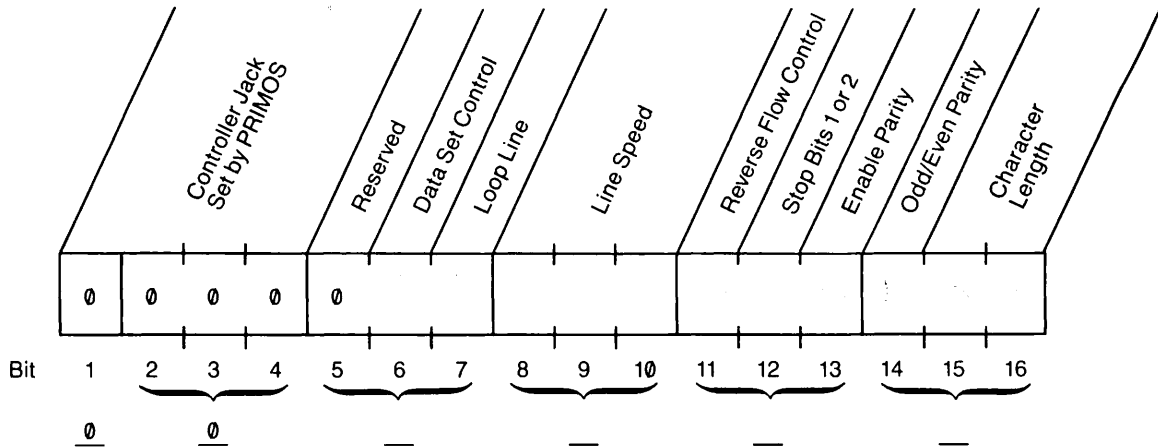
Conversion Table

BINARY	000	001	010	011	100	101	110	111
OCTAL	0	1	2	3	4	5	6	7

Building the Lword
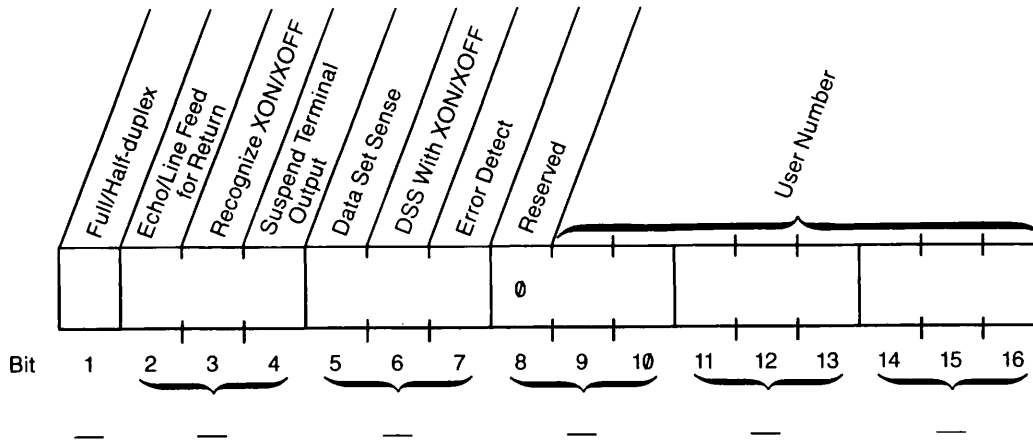
FIGURE A-1.   AMLC Octal Bitstring Templates

The *config_word* is a composite of sixteen bits, each one having its own meaning as described below. Use the template in Figure A-1 to construct a *config_word* beginning at the left, with bit 1.

Bit Number	*config_word Meaning*
1,2,3,4	Line number on controller. Set by PRIMOS. Leave as 0.
5	Reserved. Set to 0.
6	0: Data set control off. 1: Data set control on, required for modems and port selectors, ignored by terminals. (Default)
7	0: Do not loop line. (Default) 1: Loop line (used only for testing purposes).
8,9,10	Asynchronous line speed (bits per second).

000	110
001	134.5
010	300
011	1200 (Default)
100	Programmable clock. (See AMLCLK directive.)
101	75 *
110	150 *
111	1800 *

The values marked with an asterisk are the defaults supplied by Prime. If you have assigned other baud rates with the ASYNC JUMPER directive or with hardware jumper cables, your values are different.

11	0: Disable Reverse Flow Control. (Default) 1: Enable Reverse Flow Control.
12	0: 1 stop bit. (Default) 1: 2 stop bits.
13	0: Enable parity. 1: Disable parity. (Default)
14	0: Odd parity. (Default) 1: Even parity.
15,16	Character length:

00	5 bits
10	6 bits
01	7 bits
11	8 bits (Default)

The *lword*, like the *config_word*, is a composite of sixteen bits, each one having its own meaning as described below.   Use the template in Figure A-1 to construct an *lword* beginning at the left, with bit 1.

Bit Number			*lword Meaning*
1	Reset	(0)	Full-duplex. (Default)
	Set	(1)	Half-duplex.
2	Reset	(0)	Echo *LINE FEED* for *RETURN*. (Default)
	Set	(1)	Do not echo *LINE FEED* for *RETURN* (meaningful only if bit 1 is set).
3	Reset	(0)	Do not recognize XOFF and XON.
	Set	(1)	Recognize XOFF (CONTROL-S, ASCII 223) and XON (CONTROL-Q, ASCII 221).
4	Reset	(0)	Terminal output not currently blocked.
	Set	(1)	Terminal output currently suspended (XOFF seen).
5	Set	(1)	Use data set sense.
6	Reset	(0)	Examined only if bit 5 is set. Toggles between XOFF when data set sense is on, and XON when data set sense is off.
	Set	(1)	Toggles between XOFF when data set sense is off, and XON when data set sense is on.
7	Set	(1)	Enable error detection, send NAK, ASCII 225 if parity error or an overflow is sensed.
8	---		Reserved. Set to 0.
9-16	---		User number (normally line number plus 2). 0 means the line is assignable. (The system's default formula for determining the user number is *usernumber = linenumber + 2*.) The user number is displayed with the STATUS USER command described in Chapter 4, Configuring Asynchronous Lines.

Caution

If you do not follow the system's default formula for the user number, two separate AMLBUF directives are required to set the size of the DMQ buffer and the input and output ring buffers for terminal users.

NTS Considerations

NTS (Network Terminal Service) is a separately priced product that permits remote terminals and hosts to communicate with a Prime host over a LAN300, the Prime IEEE 802.3 local area network. A LAN Host Controller (LHC) installed in the Prime host provides both connection services and IEEE 802.3 network management. Clusters of remote terminals, and other asynchronous devices are connected to a LAN Terminal Server (LTS). The LTS provides the interface between these devices and the IEEE 802.3 network.

These asynchronous NTS lines can be configured with the SET_ASYNC command, as well as the AMLC command. For more information on the SET_ASYNC command, see Chapter 4, Configuring Asynchronous Lines. NTS commands are discussed in Chapter 7.

When you use the AMLC command to configure the asynchronous lines attached to an LTS, certain restrictions apply.

- You must start NTS before you can configure NTS lines.

- NTS line numbers range from 2000_8 (1024) through 2377_8 (1279).

- NTS supports six standard protocols: TTY, TRAN, TT8BIT, TTYUPC, TTY8, and TTY8HS.

- NTS users are allowed twenty seconds to complete each ASSIGN or UNASSIGN command.

- The Network Administrator must map NTS lines with the NTS_ASSOCIATE command before they can be assigned.

- NTS ignores Data Set Control.

- NTS supports Reverse Flow Control. Set RFC thresholds with either the NTSBUF or NTSABF configuration directives.

- User numbers for NTS terminals range from (NTSUS1) through (NTSUS1 + NTSUSR - 1).

- User numbers for NTS assignable lines range from (TRMUSR + NAMLC + NTSASL) through (TRMUSR + NAMLC + NTSASL).

NTS Protocols

NTS supports six standard protocols: TTY, TRAN, TT8BIT, TTYUPC, TTY8, and TTY8HS. For NTS login lines, the LTS sends the appropriate protocol to PRIMOS when the connection is established. The previous section, Standard Protocols, describes each protocol in detail.

The protocols TTYHS, TRANHS, or TTYHUP are intended for older model 5054 DMT AMLC controller boards. When you specify one of these protocols, NTS automatically converts the protocol to its equivalent: TTY, TRAN, or TTYUP. An error message is not displayed.

Several protocols are not supported. If you specify TTYNOP or ASD for a line attached to an LTS, no action is taken and an error message is displayed on the screen.

NTS User Numbers

The user number associates a buffer with a physical line. The default user number for local lines is the physical line number plus two. For local assignable lines the user number is always zero. As shown below, user numbers for NTS lines are higher than those for local and PRIMENET remote login lines. Therefore, existing AMLBUF and REMBUF directives continue to work without changes, regardless of whether or not NTS is configured. The range of acceptable values for user number is as follows:

Line Type	*User Number Range*
Local Terminal Lines	1 to NTUSR
Remote Terminal Lines	RUSR1 to RUSR1 + NTUSR - 1
NTS Terminal Lines	NTSUS1 to NTSUS1 + NTSUSR - 1
Local Assignable Lines	TRMUSR + 1 to TRMUSR + NAMLC
NTS Assignable Lines	TRMUSR + NAMLC + 1 to TRMUSR + NAMLC + NTSASL

$$\text{where TRMUSR = NTUSR + NRUSR + NTSUSR}$$
$$\text{RUSR1 = NTUSR + 1}$$
$$\text{NTSUS1 = RUSR1 + NRUSR}$$

Caution

NTS lines can use only those user numbers within the acceptable range given above. Do not attempt to swap buffers by substituting an NTS user number for a local or remote login line.

▶ ASSIGN AMLC

The octal ASSIGN AMLC command was replaced by the decimal ASSIGN ASYNC command at PRIMOS Rev. 21.0. For more information on the new command format, see Chapter 5, Assignable Asynchronous Lines.

The ASSIGN AMLC command configures and assigns an asynchronous communication line. You also can assign a previously mapped path between devices.

The ASSIGN AMLC command associates the assigned line with the first free pair of input and output ring buffers from the pool of buffers reserved for assignable lines. A line can be assigned only if it is configured as an assignable line in your PRIMOS.COMI file and the line is not currently assigned to another user or process.

Before you can assign an NTS line, you must map it with the NTS_ASSOCIATE command. NTS allows a twenty second time window for each connection request before terminating the ASSIGN command.

After the system is running, you can assign the line from your terminal by issuing the following command:

<u>AS</u>SIGN   AMLC   [protocol]   line   [config_word]

Argument	Description
protocol	Defines the format and relative timing of data exchanged over the line, where *protocol* can be one of the following: ASD, TRAN, TTY, TT8BIT, TTYUPC, TTY8, TTY8HS, TTYHS, TRANHS, or TTYHUP. NTS lines support TTY, TRAN, TT8BIT, and TTYUPC
line	Specifies the octal line number. The maximum values are 377_8 (255) for local lines and 2377_8 (1279) for NTS lines.
config_word	An octal bit string that defines the configuration parameters. For detailed information see the section, Constructing Octal Bitstrings, in this appendix. The four most commonly used *config_words* and the baud rates associated with them are as follows:

config_word	Baud Rate
2033	110 Baud (uses 2 stop bits)
2213	300 Baud
2313	1200 Baud (Default)
2413	Programmable clock value (clock default is 9600 Baud)

Note

The ASSIGN AMLC command cannot be issued twice in succession to perform an implicit UNASSIGN. To change a line from an assigned line to a login line, you must first issue the UNASSIGN AMLC command.

▶ **UNASSIGN AMLC**

The octal UNASSIGN AMLC command was replaced by the decimal UNASSIGN ASYNC command at PRIMOS Rev. 21.0. For more information on the new command format, see Chapter 5, Assignable Asynchronous Lines.

The UNASSIGN AMLC command releases an asynchronous communication line and makes it available for another user or process.

Use the UNASSIGN command to break a permanent connection over an NTS line. NTS allows a twenty second time window for each disconnect request before terminating an UNASSIGN command. After issuing the UNASSIGN command for an NTS line, issue the NTS_UNASSOCIATE command.

The format of the UNASSIGN AMLC command is as follows:

UNASSIGN AMLC number

Argument   *Description*

number   Specifies the octal line number. The maximum values are 377_8 (255) for local lines and 2377_8 (1279) for NTS lines.

Remember, you can unassign only those lines that you have assigned. If necessary, the operator or System Administrator has the rights to unassign any line or device from the supervisor terminal, regardless of who assigned it or from which terminal it was assigned.

AMLC Error Messages

The AMLC command checks for errors while parsing the command line. Missing, contradictory, or incorrect arguments generate an error message and no action is taken on the command. Error messages are reported to the process that invoked the AMLC command, usually the supervisor terminal. The AMLC error messages are listed alphabetically below.

`Buffer number <buffer number> is already in use by line <n>.  (AMLC)`
    An attempt was made to assign a single buffer *buffer number* to more than one line.

`Buffer <buffer number> is a remote login buffer. (AMLC)`
    An attempt was made to use a remote buffer *buffer number* for a local line.

`Buffer number <n> is out of range; the acceptable range for <line type> buffers is`
`<x to y>.  (AMLC)`
    The user number *n* specified in the lword is out of range for the given line. Where *line type* can be Local Terminal, Local Assigned, NTS Terminal, or NTS Assigned and *x to y* is the acceptable range of values for that line type.

`Cannot enable auto speed detect on an assignable line. (AMLC)`
    An attempt was made to specify the ASD protocol on an assignable line.

`Cannot refer to an NTS line when NTS is not started. (AMLC)`
    An AMLC command was issued to configure an NTS line when NTS was not started.

`Cannot swap terminal/assigned buffers while an NTS line is connected.  (AMLC)`
    The user number specified in the lword attempted to substitute an NTS assigned line's buffer for an NTS terminal buffer while the line is connected, or vice versa.

`Cannot use an assignable line buffer <buffer number> on a user login line. (AMLC)`
    An attempt was made to allocate a buffer *buffer number* from the pool reserved for assignable lines to a login line.

`Device output queue full.  (AMLC)`
    The line control output queue cannot accommodate the line configuration block generated by the AMLC command. This error message may indicate a temporary failure of PRIMOS to communicate with the LHC controller, or a problem with the LHC controller itself. No action is taken on the command attempt. However, if the command is reissued, it may succeed.

`Invalid ASD sample speed specified <speed>.` (AMLC)
> The baud rate given in the config__word is not a valid ASD sample speed.

`Line is already assigned to another user <user number>, must be unassigned first.` (AMLC)
> An attempt was made to convert a currently assigned line to a login line or assign a line that is already being used by another user.

`Line <n> now supports user login, user number is <user number>.` (AMLC)
> This message is displayed after an assignable line *n* is reconfigured to be a login line with the given *user number*.

`Line number is required.` (AMLC)
> A valid line number was not specified in the command line.

`Line number <n> is out of range; acceptable range is 0 to 377`$_8$`.` (AMLC)
> When the octal line number *n* supplied in the command line is not valid.

`Line number <n> is out of range; acceptable range is 0 to 377`$_8$ `and 2000`$_8$ `to 2377`$_8$`.` (AMLC)
> The octal line number *n* supplied in the command line is not valid and NTS is running.

`No XCB available for request.` (AMLC)
> This message usually indicates a software failure, either in PRIMOS or the LHC. Contact your Customer Support Center.

`The <protocol name> protocol is not supported on NTS lines.` (AMLC)
> An unsupported protocol was supplied in the command line, where the invalid *protocol name* can be TTYNOP, or ASD.

`System console command only.` (AMLC)
> An AMLC command was issued from a terminal other than the supervisor terminal (formerly referred to as the system console).

`WARNING: Changing the speed of the AMLC Clock Line <n> to greater than 300 bps.` (AMLC)
> The line <n>, the last line of the last AMLC controller board, was configured for a speed greater than 300 bps. This can result in severe performance degradation.

`WARNING: Line number <n> does not exist on this system.` (AMLC)
> An attempt was made to configure a line that does not physically exist.

B

DETERMINING PHYSICAL LINE NUMBERS

This appendix describes the procedure for tracing your asynchronous communication lines back to the controller. It explains how to calculate a physical line number from the controller's device address, offset, and jack number.

Determining Line Numbers

Use the following procedure to find the line number for a particular asynchronous line:

1. Use the STATUS COMM command to determine the device addresses for all the AMLC and ICS controller boards in the system.

2. Trace the target line to the controller into which it is plugged. The line is connected to a cable assembly, which consists of four cables. The four cables terminate at one end in a single connector, marked C1. Each of the other ends has a nine-pin connector labeled J1 to J4. The cable assembly, in turn, is plugged into one of one, two, or four ports on the target controller.

3. Determine the type of the target controller (AMLC, ICS1, ICS2, or ICS3).

4. Find out the device address of the target controller. Examples are 10_8, 11_8, 36_8, and 54_8.

5. Determine into which port the C1 cable connector is plugged.

6. Find out the jack number into which the target line is plugged. The four cables on a given cable assembly are labeled J1, J2, J3, and J4. The labels are located near the point at which the target line connects to the cable.

7. Use one of the following two procedures to determine the target line number. For ICS controllers, see the section entitled Lines Attached to ICS Controllers. For AMLC controllers, see the section entitled Lines Attached to AMLC Controllers.

Lines Attached to ICS Controllers

To determine the line number for a line connected to an ICS controller, use the following procedure:

1. Determine the offset of the first ICS controller on the system.

2. Determine the offset of the target controller.

3. Determine the offset of the line on the target controller.

4. Add the three numbers obtained from steps 1-3. The line number is the sum of these three numbers.

This procedure is explained in the following paragraphs. For more information on ICS controllers, see the *ICS User's Guide.*

Finding the Offset of the First ICS Controller: AMLC controllers are always configured before ICS controllers. Therefore, the offset of the first ICS controller depends on whether the system has AMLC controllers.

● If the system has no AMLC boards, the offset of the first ICS controller is zero.

● If the system has AMLCs, the offset of the first ICS controller is the highest available AMLC line number rounded up to the next modulus 16 boundary. The offset is always represented in octal, is always evenly divisible by 16 (20_8), and always ends in 0. For example, offsets can be 0, 20_8, 40_8, 60_8, and 100_8.

Finding the Offset of the Target Controller: The offset of the target ICS controller depends on the configuration of the lines and controllers. However, the offset for AMLC controllers depends entirely on the device address.

1. Use the STATUS COMM command to find the device addresses of all the ICS controllers. The address priority for all ICS controllers is 10_8, 11_8, 36_8, and 37_8. ICS2 and ICS3 controllers are usually assigned a device address of 10_8 or 11_8 by the manufacturer. When you have ICS2 or ICS3 controllers as well as an ICS1, the ICS1 controller has a lower priority and is assigned a device address of 36_8 or 37_8. If you have only ICS1 controllers, all device addresses (36_8, 37_8, 10_8, or 11_8) are valid.

2. Check off the device addresses of the ICS controllers on the list in Table B-1. Do not, however, check off AMLC device addresses on the list.

3. The ICS controller indicated by the first check in the list has a target offset of 0. The target offset of an additional ICS controller depends on the number of lines configured for those above it on the list.

Number of Lines Per Controller: The number of lines on an ICS controller depends on the model; an ICS1 controller is always configured for eight asynchronous lines. ICS2 and ICS3 controllers support four asynchronous lines for every Line Adapter Card (LAC) in the card cage. Use the following procedure to determine how many lines are present.

1. Determine how many asynchronous LACs are in the controller's card cage.

2. If the number from step 1 is odd, add one to it.

3. Multiply the result by 4. Write this number in the *Number of Lines* column of the controller checklist in Table B-1.

4. Add the number of lines configured for the first ICS to its offset (zero). The result gives the offset of the second ICS. If the system has a third ICS, its offset is the sum of the number of lines configured for the second ICS and the offset of the second ICS, and so on.

TABLE B-1.   ICS Controller Checklist

Present	Address	Offset	Number of Lines
[ ]	10_8	[ ]	[ ]
[ ]	11_8	[ ]	[ ]
[ ]	36_8	[ ]	[ ]
[ ]	37_8	[ ]	[ ]
[ ]	56_8	[ ]	[ ]
[ ]	51_8	[ ]	[ ]
[ ]	50_8	[ ]	[ ]
[ ]	32_8	[ ]	[ ]
[ ]	17_8	[ ]	[ ]
[ ]	16_8	[ ]	[ ]
[ ]	15_8	[ ]	[ ]
[ ]	35_8	[ ]	[ ]
[ ]	52_8	[ ]	[ ]
[ ]	53_8	[ ]	[ ]
[ ]	54_8	[ ]	[ ]

Finding the Line Number on the Target Controller: The line number is determined differently for each model of ICS controller.

For an ICS1 controller, the line number depends on

● The port to which the line is connected on the ICS1 controller

● The cable connector into which the line is plugged

An ICS1 controller has three ports.   The leftmost port is for synchronous communication only, so it is not discussed here. The center and rightmost ports are the asynchronous ports. The rightmost port contains controller line numbers 0 to 3, and the middle port has board line numbers 4 to 7. (Note that this is the reverse of the AMLC board.)

Use the following list to determine the ICS1 board line number.

Port	*J1*	*J2*	*J3*	*J4*
Rightmost	0	1	2	3
Middle	4	5	6	7

For ICS2 and ICS3 controllers, the board line number depends on

● The LAC to which the line is connected on the ICS2 or ICS3

● The jack number of that particular line on the LAC

ICS2 and ICS3 controllers support up to 16 LACs.   Figure B-1 shows the jacks and cable connections for ICS2 and ICS3 asynchronous LACs.   The LACs are numbered from right to left as viewed from the back of the ICS2.   The rightmost cable leads to a buffer card. The number of other cables depends on how many LACs are on the controller.   Each of the next eight cables leads to a LAC. If the system has more than eight LACs, the next cable leads to a buffer card, and the following cables lead to LACs.

Determine how many LACs are to the right of the LAC to which the line is connected, and to which of the four jacks the line is connected.   Now multiply the number of LACS by four, add the jack number, and subtract one.   The result is the board line number.

Calculating Line Numbers: To calculate the line number, add the ICS line offset, the target offset, and the board line number.   This value is the line number in octal.   You can convert this number to a decimal number with the following PRIMOS command for octal-to-decimal conversions:

```
TYPE [OCTAL octal-number]
```
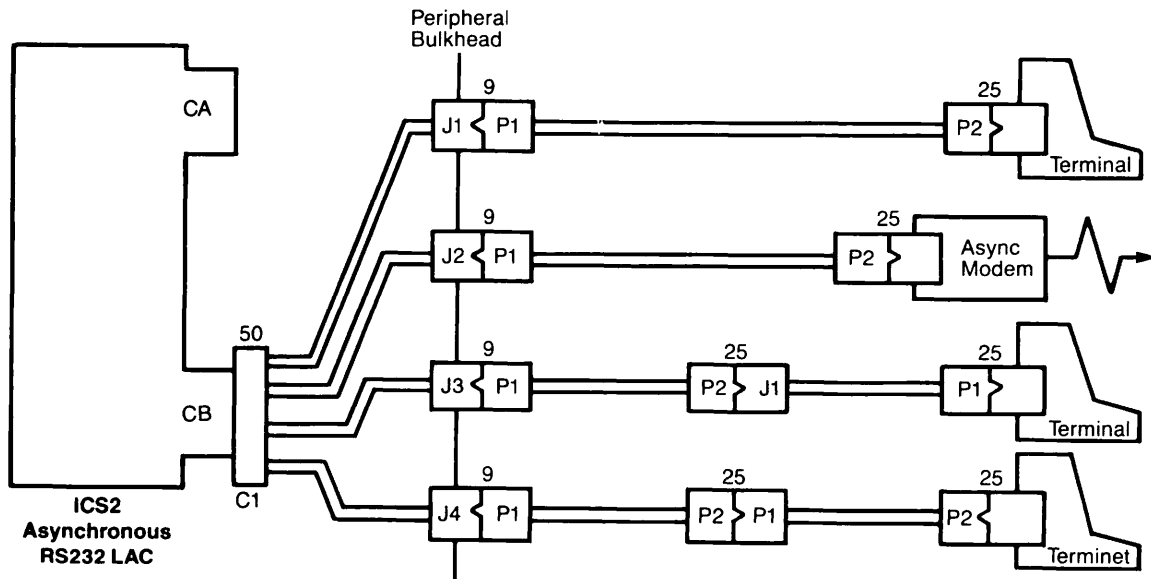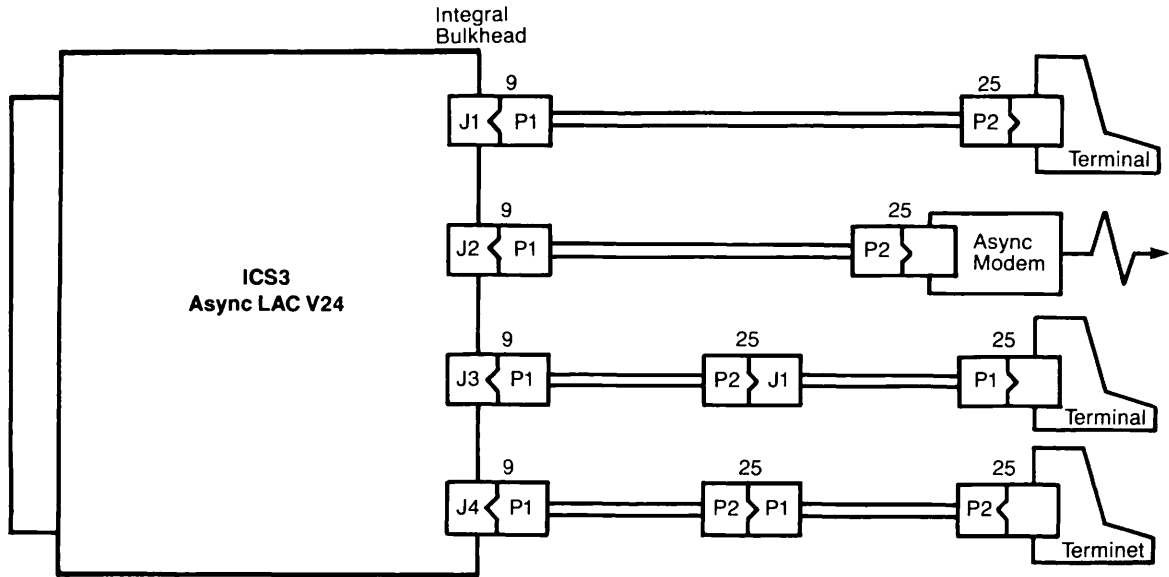
FIGURE B-1. Cable Connections for ICS2 and ICS3 Asynchronous LACs

Lines Attached to AMLC Controllers

A system can have a maximum of eight AMLC boards. Each board has a maximum of four ports (C, D, E, F), proceeding from left to right as viewed from the rear of the CPU. Line numbers can be calculated with the formula given in Table B-2. Tables B-3 and B-4 list the line assignments for four AMLC controllers.

TABLE B-2. Determining AMLC Line Numbers

AMLC Board	Value of x =	Device Address
1	0	54_8
2	1	53_8
3	2	52_8
4	3	35_8
5	4	15_8
6	5	16_8
7	6	17_8
8	7	32_8

AMLC Port	Value of y =	Jack Number	Value of z =
C	0	J1	1
D	1	J2	2
E	2	J3	3
F	3	J4	4

To determine:	*Use the default formula:*
1) Physical line number	$16(x) + 4(y) + (z-1) = n$
2) SET_ASYNC command *line* argument	Convert the physical line number *n* calculated above to decimal with the PRIMOS command TYPE [OCTAL n]

TABLE B-3.   AMLC Line Number Assignment

First AMLC Board -- Device Address 54_8

Port	Jack	Line #	User #
C	A1-J1	1 (0_8)	2
C	A1-J2	2 (1_8)	3
C	A1-J3	3 (2_8)	4
C	A1-J4	4 (3_8)	5
D	A2-J1	5 (4_8)	6
D	A2-J2	6 (5_8)	7
D	A2-J3	7 (6_8)	8
D	A2-J4	8*(7_8)	9
E	A3-J1	9 (10_8)	10
E	A3-J2	10 (11_8)	11
E	A3-J3	11 (12_8)	12
E	A3-J4	12 (13_8)	13
F	A4-J1	13 (14_8)	14
F	A4-J2	14 (15_8)	15
F	A4-J3	15 (16_8)	16
F	A4-J4	16*(17_8)	17

Second AMLC Board -- Device Address 53_8

Port	Jack	Line #	User #
C	A1-J1	17 (20_8)	18
C	A1-J2	18 (21_8)	19
C	A1-J3	19 (22_8)	20
C	A1-J4	20 (23_8)	21
D	A2-J1	21 (24_8)	22
D	A2-J2	22 (25_8)	23
D	A2-J3	23 (26_8)	24
D	A2-J4	24*(27_8)	25
E	A7-J1	25 (30_8)	26
E	A7-J2	26 (31_8)	27
E	A7-J3	27 (32_8)	28
E	A7-J4	28 (33_8)	29
F	A8-J1	29 (34_8)	30
F	A8-J2	30 (35_8)	31
F	A8-J3	31 (36_8)	32
F	A8-J4	32*(37_8)	33

*If this is the last available AMLC line,
it establishes the AMLC I/O interrrupt rate.
Set the speed of this line to 110 or 300 baud.

TABLE B-4.   AMLC Line Number Assignment

Third AMLC Board -- Device Address 52_8

Port	Jack	Line #	User #
C	A9-J1	33 (40_8)	34
C	J103	34 (41_8)	35
C	J102	35 (42_8)	36
C	J101	36 (43_8)	37
D	J100	37 (44_8)	38
D	J99	38 (45_8)	39
D	J98	39 (46_8)	40
D	J97	40*(47_8)	41
E	J96	41 (50_8)	42
E	J95	42 (51_8)	43
E	J94	43 (52_8)	44
E	J105	44 (53_8)	45
F	J115	45 (54_8)	46
F	J114	46 (55_8)	47
F	J113	47 (56_8)	48
F	J112	48*(57_8)	49

Fourth AMLC Board -- Device Address 35_8

Port	Jack	Line #	User #
C	J111	49 (60_8)	50
C	J110	50 (61_8)	51
C	J109	51 (62_8)	52
C	J108	52 (63_8)	53
D	J107	53 (64_8)	54
D	J106	54 (65_8)	55
D	J116	55 (66_8)	56
D	J117	56*(67_8)	57
E	J125	57 (70_8)	58
E	J124	58 (71_8)	59
E	J123	59 (72_8)	60
E	J122	60 (73_8)	61
F	J121	61 (74_8)	62
F	J120	62 (75_8)	63
F	J119	63 (76_8)	64
F	J118	64*(77_8)	65

*If this is the last available AMLC line,
it establishes the AMLC I/O interrrupt rate.
Set the speed of this line to 110 or 300 baud.

Example of Line Number Calculation

Suppose a system has the following controllers:

- One AMLC controller

- One ICS2 controller, at device address 10_8, containing seven Line Adapter Cards

- Two ICS1 controllers, at device addresses 11_8 and 36_8

To determine the number of the line connected to cable connector J3 on the middle port of the ICS1 at device address 36_8, proceed as follows:

1. Find the offset of the first ICS controller. Because the system has only one AMLC, this offset is 20_8.

2. Use the STATUS COMM command to check the device addresses of the ICS controllers, and check these off on the ICS Controller Checklist. (See the sample under step 5 below.)

3. Determine the number of lines that are allocated for the ICS2. Because there are 7 LACs, add 1 and multiply by 4. The result is 32 in decimal. Converted to octal, the answer is 40_8.

4. Find the offset of the ICS1 at device address 11_8. Compute this by adding the offset of the first ICS (0) to the number of lines, 40_8.

5. Find the offset of the ICS1 at device address 36_8. This is the sum of the offset of the second ICS (40_8) and the number of asynchronous lines (10_8). (An ICS1 is always configured for 10_8 lines.) Fill in the appropriate information on your checklist as shown in the following example.

Completed Sample ICS Controller Checklist

Present	*Address*	*Offset*	*Number of Lines*
[x]	10_8	[0_8]	[40_8]
[x]	11_8	[40_8]	[10_8]
[x]	36_8	[50_8]	[10_8]

6. Find the board line number. In the preceding list of ICS1 board line numbers, the line connected to J3 on the middle port is line number 6.

7. Add the ICS line offset from step 1 (20_8), the offset of the target ICS1 from step 4 (50_8), and the board line number from step 6 (6_8). The result is 76_8.

8. Use the PRIMOS command TYPE [OCTAL 76] to convert the answer to decimal and display it on the screen. The result is 62. The line number is therefore 62.

INDEX

INDEX

SURVEYS

READER RESPONSE FORM

SYSTEM ADMINISTRATOR'S GUIDE, VOLUME II
DOC10132-1LA

Your feedback will help us continue to improve the quality, accuracy, and organization of our publications.

1. How do you rate this document for overall usefulness?

   □ *excellent*   □ *very good*   □ *good*   □ *fair*   □ *poor*

2. What features of this manual did you find most useful?

   _____
   _____
   _____
   _____
   _____
   _____

3. What faults or errors in this manual gave you problems?

   _____
   _____
   _____
   _____
   _____
   _____

4. How does this manual compare to equivalent manuals produced by other computer companies?

   □ *Much better*   □ *Slightly better*   □ *About the same*
   □ *Much worse*   □ *Slightly worse*   □ *Can't judge*
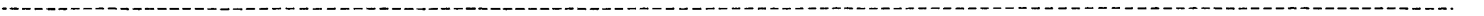
5. Which other companies' manuals have you read?

   _____
   _____

Name:_____ Position:_____

Company:_____

Address:_____

_____

_____ Postal Code:_____