ACC HDLC/LAP-B (ABM) User's Guide

Edition 9

HP 9000 Networking

Manufacturing Part Number: Z7487-90001

E0400

Legal Notices

Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be held liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

The information in this document is subject to change without notice.

Warranty. A copy of the specific warranty terms applicable to your Hewlett- Packard product and replacement parts can be obtained from your local Sales and Service Office.

Restricted Rights Legend. Use, duplication or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 for DOD agencies, and subparagraphs (c) (1) and (c) (2) of the Commercial Computer Software Restricted Rights clause at FAR 52.227-19 for other agencies.

HEWLETT-PACKARD COMPANY 3000 Hanover Street Palo Alto, California 94304 U.S.A.

Use of this manual and flexible disk(s) or tape cartridge(s) supplied for this pack is restricted to this product only. Additional copies of the programs may be made for security and back-up purposes only. Resale of the programs in their present form or with alterations, is expressly prohibited.

Printing History

The manual publishing date and part number indicate its current edition. The publishing date will change when a new edition is published. Minor changes may be made without changing the publishing date. The manual part number will change when extensive changes are made.

Manual updates may be issued between editions to correct errors or document product changes. To ensure that you receive the updated or new editions, you should subscribe to the appropriate product support service. See your HP sales representative for details.

First Edition	**December 1992**	**Release 1.0**
Second Edition	September 1993	Release 2.0
Third Edition	March 1994	Release 2.1
Fourth Edition	April 1995	Release 2.2
Fifth Edition	September 1995	Release B.02.20 (HP-UX 10.01)
Sixth Edition	November 1996	Release B.02.22 (HP-UX 10.10)
Seventh Edition	August 1997	Release B.02.39 (HP-UX 10.10) Release B.02.40 (HP-UX 10.20)
Eighth Edition	October 1998	Release B.03.01 (HP-UX 11.0)
Ninth Edition	**April 2000**	**Release B.03.10 (HP-UX 11.0)**

Preface

The HDLC/LAP-B (ABM) Protocol product is used in conjunction with Hewlett-Packard's Multiprotocol Advanced Communications Controller (ACC) multiplexer products.

This manual applies to the ACC Protocols for Application Developers product.

This manual explains installing the HDLC/LAP-B (ABM) protocol module and configuring it into the Multiprotocol ACC product.

Manual Organization

This manual contains the following chapters and appendices:

Chapter 1	Overview - presents an overview of the features provided by the HDLC/LAP-B (ABM) Protocol product.
Chapter 2	Software Installation and Verification - describes how to install the protocol module software. It describes how to verify that the protocol software is correctly installed and is functional.
Chapter 3	Using HDLC/LAP-B (ABM) Protocol - contains information about using the HDLC/LAP-B (ABM) protocol.
Chapter 4	Protocol Specific Configuration - provides HDLC/LAP-B (ABM) specific configuration information.
Appendix A	Sample Configuration Files - details a listing of the sample configuration files provided with this product.

Related Documentation

The documentation available for the Multiprotocol ACC family of products includes the following hardware and software manuals:

Hardware Manuals

- 8 Channel HP-PB *ACC Multiplexer Hardware Installation and Reference Manual*
- 8 Channel *EISA ACC Multiplexer Hardware Installation and Reference Manual*
- 8 Channel *PCI ACC Multiplexer Hardware Installation and Reference Manual*
- 2 Channel (HP-PB) *ACC Multiplexer Hardware Installation and Reference Manual*
- 4-Channel T1/E1 (HP-PB) *ACC Multiplexer Hardware Installation and Reference Manual*

Software Manuals

- *ACC Installation and Configuration Guide*
- *ACC Utilities Reference Guide*
- *ACC Programmer's Reference Guide*
- *ACC Error Guide*
- *HDLC Frame Protocol User's Guide*
- *ACC X.25 Protocol User's Guide*
- *ACC X.25/ISDN Data Analyzer User's Guide*
- *ACC HDLC/LAP-B (ABM) Protocol User's Guide*
- *ACC HDLC/LAP-D Protocol User's Guide*
- *HDLC-NRM (SDLC) Protocol User's Guide*
- *X.25/ACC Update Guide*
- *X.25/9000 User's Guide*
- *X.25/9000 Programmer's Guide*

Contents

Contents

1 Overview

Introduction

The HDLC/LAP-B (ABM) Protocol product is the Data Link or Level 2 layer used in conjunction with the Advanced Communications Controller (ACC) product family. The ITU (CCITT) recommends the Link Access Procedure Balanced (LAPB) protocol for frame-level information transfers on balanced data links (two data stations) in X.25 packet-switching networks. This manual provides installation and configuration information that is specific to the HDLC/LAP-B protocol and the Asynchronous Balanced Mode (ABM) of operation.

Product Features

This software provides Level 2 interface operation within the BAC class of procedures recommended by ITU-T (CCITT). As described in the standards cited below, BAC (**B**alanced operation **A**synchronous balanced mode **C**lass) applies to point-to-point configurations involving two *combined stations* on a logical data link. Unlike *primary/secondary stations*, these may have an equal (balanced) stake in the responsibility for data link management.

NOTE

For Level 2 interface operation within the unbalanced class of procedures (UNC, for **U**nbalanced operation **N**ormal response mode **C**lass), Hewlett-Packard provides a separate ACC accessory product: The HDLC NRM Protocol product. (The HDLC NRM Protocol accessory includes an SDLC loop mode option.)

The LAP-B (ABM) protocol is defined in ISO standards 3309, 4335, and 6256 as follows:

- Standard 3309 pertaining to 'Data communications - High level data link control procedures - Frame structure';
- Standard 4335 pertaining to 'Data communications - High level data link control procedures - Elements of procedures', and
- Standard 6256 pertaining to 'Data communications - HDLC balanced class of procedures'.

All were drawn up by ISO/TC 97, Computers and information processing.

Supported Devices

Any device which conforms to the published HDLC/LAP-B (ABM) standards will be supported by this implementation.

Modes of Operation

As recommended by ITU-T, the BAC class of procedure involves two combined stations operating in the Asynchronous Balanced Mode in either two-way alternate or two-way simultaneous (half-duplex or duplex) point-to-point configuration. Combined stations are characterized by the ability to send both commands and responses, receive both commands and responses, and provide for data link error recovery.

The protocol optionally provides the ability to use extended sequence numbering (modulus 128). Extended sequence numbering is selected by configuration options. Normal operation uses modulus 8 sequence numbering.

Data Rates

This protocol can be used at data rates up to 76.8k baud for 8-channel ACC multiplexer cards using the HP-PB standard backplane or128k baud for 8-channel cards installed in the EISA standard backplane. The HDLC LAP-B (ABM) Protocol product can also be used at higher rates for high-speed ACC multiplexer cards, such as the 2-Channel HP-PB, 8-Channel PCI, and the 4-channel T1/E1 products with 2.048M baud capability.

Unformatted Frame or “Pass-through” Mode

This mode is no longer available with the HDLC/LAP-B product.

The “Level 1" protocol can be used to provide this functionality. Refer to the *Level 1- HDLC Framing Protocol User's Guide* for details.

Files Provided

The HDLC/LAP-B (ABM) Protocol product provides a pre-loaded firmware file that can be used directly in the ACC network configuration files.

The HDLC/LAP-B (ABM) protocol may also be used on the same ACC Multiplexer as other protocols. If this is the case, a new firmware download (`.zabs`) file can be built to contain all of the protocols that will be used on that Mux card. This product provides the relocatable firmware files and a sample Link command (`.zlnk`) file for use in this case.

The files provided with the HDLC/ABM product are listed in the following table.

Table 1-1 Provided Files

Downloadable	
/opt/acc/<card-type>/hdlc.zabs	This is a linked file that contains the (***.zrel**) relocatables. It can be referenced directly from the network configuration file.
Relocatable Firmware Files	
For the 2-channel and 8-channel (EISA and HP-PB only) ACC cards, the relocatable firmware file provided is:	
/opt/acc/protocol/hdlcabm.zrel	Implements the Link Access Procedure (Balanced) protocol and interfaces the LAP-B link-level code to the application program set as receiver for the LAP-B point-to-point ZLU.
Other Files	
/opt/acc/<card-type>/hdlc.zlnk	This link file is used by the `zlink` program to create the `hdlc.zabs` file.
/opt/acc/cfg/lapb_sample.answ	This is a sample network configuration file.

References

For information on installing the ACC product, how to Link the relocatable firmware files, and how to start up the ACC Subsystem, refer to the *ACC Installation and Configuration Guide*.

For information on using the ZCOM **A**pplication Programmatic Interface (API), refer to the *ACC Programmer's Reference Guide*.

For information on using the utilities related to the ACC products, refer to the *ACC Utilities Reference Guide*.

For information on using the X.25 protocol on ACC products with applications that use the ZCOM API, refer to the *ACC X.25 Protocol User's Guide*.

For information on using the X.25 protocol with applications that use the X.25/9000 Product API's, refer to the *X.25/ACC Installation and Configuration Guide* and the *X.25/9000 Programmer's Guide*.

2 Software Installation and Verification

Introduction

The HDLC/LAP-B (ABM) protocol is included in the ACC Developer Software Suite product. This chapter describes how to install the product using the `swinstall` utility. It also describes how to verify that the protocol has been successfully installed and is functional.

Software Removal

Step 1. If a previous version of the HDLC/LAP-B protocol product is already on the HP-UX system, it should first be removed. It is very important to follow the instructions provided with the software that is currently installed in the system to shutdown and remove that software. This procedure may change between major releases of the product. Log in to the system as superuser.

Step 2. Terminate the ACC Mux Subsystem by typing:

```
% zmasterd stop

% zmasterd kill
```

Step 3. Run `swremove` to remove the HDLC/LAP-B protocol product.

```
% swremove
```

NOTE The `swremove` will fail if any ACC daemon is still active. Terminate any active ACC daemon indicated in the log file and try again.

Any of the ACC related products can be removed by selecting them and then marking them for removal. (Both the terminal interface and the GUI interface for swremove behave the same; only the means for selecting items differs.)

Step 4. Select the HDLC/LAP-B protocol product (and any other items to be removed) in the “Software Selection Window” and mark for removal, using the “Mark for Remove” function under the “Actions” menu. When all items are marked, select the “Remove (analysis)” action.

Open “Logfile” to view running status of the removal process, and follow any screen prompts until the removal process is done.

Software Installation

Running Swinstall

The ACC software is read from the HP-UX Application CD-ROM, in superuser mode, using the `swinstall` utility. The steps are shown below.

Step 1. Log in to the system as “root”.

Step 2. Place the media in a local or remote CD-ROM drive.

Step 3. Run `swinstall`, which behaves the same in the GUI interface as for the terminal interface. When the “Specify Source” box opens, select “Source Depot Type” to “Local CDROM”, or for a remote CD-ROM driver, select for “Network Directory/CDROM” and set “Source Host Name...”. Select “OK”.

Step 4. Select the ACC Developer Software Suites product and any other ACC items to be installed, from the list of software bundles; open the “Actions” menu and select “Mark for Install”.

Step 5. When all items have been marked, select “Install (analysis)”; select “OK” and proceed.

The “Install (analysis)” window allows you to track summary progress of the installation process. Use the “Logfile” feature to display a detailed status, which is logged to `/var/adm/sw/swagent.log`.

Product Identification

The **what** utility can be run on `/opt/acc/<card-type>/hdlc.zabs` or `/opt/acc/protocol/hdlcabm.zrel` to show what version of the protocols is installed.

Installation Verification

A sample configuration file `/opt/acc/cfg/lapb_sample.answ` file is provided with the ACC Developer Software Suite product. This sample file can be used to verify that the HDLC/LAP-B (ABM) protocol module has been properly installed and is functional. Note that you may need to customize this file, to reflect the location of the mux card(s) and to reflect the mux ports to be used. The `lapb_sample.answ` file is shown in Appendix A , “Sample Configuration Files,” on page 55.

The steps in the verification process are listed below. It is assumed that the ACC Developer Software Suite product is installed on the system, and that there is at least one ACC Mux card installed with a cable and mux panel attached.

At least one loopback cable should be used to connect two of the ports on the mux panel. For the configuration file provided, it is assumed that the first two ports of the mux card are connected with a loopback cable, and that the end of the cable marked “Int” is connected to port 0.

Using the HDLC/LAP-B (ABM) Protocol Module

The *ACC Installation and Configuration Guide* contains the general procedure for configuring the network configuration file, creating customized downloadable firmware files (except for T1/E1 4-channel cards), and starting up the ACC subsystem with the new protocol and new configuration. To add HDLC/LAP-B (ABM) to a firmware file, see “Download Linkage” on page 59 in Appendix A of this manual.

Refer to Chapter 4 , “Protocol Specific Configuration,” in this manual for HDLC/LAP-B (ABM) specific configuration requirements when building the network configuration file.

Verification Procedure

The procedure described involves using the `zmlog`, `ttgen`, `zmntr`, and `zterm` utilities. These utilities are described in the *ACC Utilities Reference Guide*.

Step 1. Modify the interface definition statement in the `/opt/acc/cfg/lapb_sample.answ` to reflect the location of the ACC Mux cards if necessary. (Refer to Chapter 4 , "Protocol Specific Configuration," in this manual for information on how to modify this statement.) Also modify the <card-type> term for the Interface-Definition statement to "z7200a", or your card type.

Step 2. The sample file only configures the first two ports of one mux card. If desired, expand the configuration file to include additional ports.

NOTE For the test described here, the ports that will be connected together via a loopback cable must have one port configured as DTE, and the other as DCE. The mux ports must also have one set to internal clocking and the other to external clocking.

In the sample configuration file, the even ports are set to internal clocking and the odd ports are set to external clocking. Also note that for 2-port and 8-port cards the loopback cable must be connected so that the end marked "Int" is connected to the port configured for internal clocking.

Step 3. Run `ttgen` on the modified network configuration (.answ) file.

```
% ttgen -o lapb_sample.answ lapb_sample.tmem
ttgen: END$  0 Disasters, 0 Errors, 0 Warnings
%
```

Step 4. Bring up the ZCOM subsystem:

```
% zmasterd cold /opt/acc/cfg/lapb_sample.tmem
```

Monitor the appropriate `/var/opt/acc/log/*.tlog` file, e.g. `mon.tlog`, `tue.tlog`, etc., to make sure the ACC ZCOM subsystem comes up. The following `zmlog` messages should be seen:

```
------------------------------------------------------------------
MON JUN 16 13:36:25 1998: ZMLOG: MESSAGE LOGGING RESUMED
------------------------------------------------------------------
13:36:11 ZNODE 00109   zcom SYSTEM DOWN, EXITING
13:36:11 ZMAST 00129   STOPPING ZMLOG DAEMON.
13:36:11 ZMON  00049   END OF zMON REQUEST, PROGRAM TERMINATED
13:36:11 ZMON  00075   zcom SYSTEM STOPPED
13:36:24 ZMAST 00101   LAUNCHED DAEMON ZMLOG, PID 1433.
13:36:24 ZMAST 00117   ZMASTERD DAEMON START RUNNING ...
13:36:24 ZMAST 00101   LAUNCHED DAEMON ZMON, PID 1434.
13:36:25 ZMON  00002   RESOURCE MANAGER (REV 1.32) FOR zcom 6.2.0.0
13:36:25 ZMON  00003   COLD START WITH: LAPB_SAMPLE.TMEM
13:36:25 ZMON  00100   CARD 0 STARTING UP ...
13:36:32 ZMON  00110   CARD 0 STARTUP SUCCESSFUL, CARD READY
13:36:32 ZMON  00020   COLD START COMPLETED, zcom SYSTEM READY
13:36:32 ZMON  00004   WAITING FOR ZMON REQUESTS ...
13:36:32 ZMAST 00101   LAUNCHED DAEMON ZNODE, PID 1455.
13:36:32 ZCOM  00165   NODE 123 IS NOW
```

Step 5. The *zmntr* utility can be used to display the original state of the HDLC link:

```
ZMNTR> tt 10 20
 ZLU# MX P:SC TERMINAL DESCRIPTION..  RX.MES TX.MES ERRORS E.RATE STATE.......
 0010 00 0:00 HDLC LINK 1                0      0      0  0.00% DISABLED
 0020 00 1:00 HDLC LINK 2                0      0      0  0.00% DISABLED
 **** ** ** MESSAGE TOTALS               0      0      0   NAN% ***********
ZMNTR> ex
#
```

Step 6. The link can now be enabled using the *zterm* utility:

```
# zterm
13:40:09 ZCOM INTERACTIVE COMMAND UTILITY
13:40:09 PRIMARY ZLU is  801
ZTERM> rc 10 20
ZTERM> cn 10 20 EN
ZTERM> rx
13:40:29 STAT CHG (UP) RSPNS FROM ZLU#00020(00123) NO ERROR DETECTED
13:40:29 STAT CHG (UP) RSPNS FROM ZLU#00010(00123) NO ERROR DETECTED

<CNTRL-C>

13:40:37 ERROR ON ZREAD: INTERRUPT OCCURS WHILE WAITING
13:40:37 MESSAGES RECEIVED         2,    MESSAGES SENT            0
13:40:37 ELAPSED TIME SECS         8,    MESSAGES/SEC             0
13:40:37 DATA IN KBytes            0,    DATA RATE (KB/S)         0
13:40:37 Sys CPU   0%, User CPU   0%,    TOTAL CPU   0%
13:40:37 MUX   0 UTILIZATION      1%
ZTERM> EX
```

Step 7. The *zmntr* utility can be used to observe the state of the link:

```
ZMNTR> tt 10 20
 ZLU# MX P:SC TERMINAL DESCRIPTION..  RX.MES TX.MES ERRORS E.RATE STATE.......
 0010 00 0:00 HDLC LINK 1                  0      0      0  0.00% DEACTIVATED
 0020 00 1:00 HDLC LINK 2                  0      0      0  0.00% DEACTIVATED
 **** ** ** MESSAGE TOTALS                 0      0      0   NAN% ************
ZMNTR> EX
```

Step 8. The link can now be brought up using the *zterm* utility:

```
# zterm
13:42:40 ZCOM INTERACTIVE COMMAND UTILITY
13:42:40 Primary ZLU is  801
ZTERM> rc 10 20
ZTERM> cn 10 20 ac  [Activate the link.]
ZTERM> ex
```

NOTE Note that no status change will be received for the activate request.

Step 9. The *zmntr* utility can be used to observe the state of the link:

```
ZMNTR> tt 10 20
ZLU# MX P:SC TERMINAL DESCRIPTION..  RX.MES TX.MES ERRORS E.RATE STATE.......
 0010 00 0:00 HDLC LINK 1                  0      0      0  0.00% Up
 0020 00 1:00 HDLC LINK 1                  0      0      0  0.00% Up
 **** ** ** Message totals                 0      0      0   .00% **
ZMNTR> ex
```

Step 10. Data can now be sent over the established link. In this example, two 50 byte messages are sent on ZLU 10 using `zsend` mode 0 (indicating that no response is desired when messages have been sent).

```
% zterm
13:44:28 ZCOM INTERACTIVE COMMAND UTILITY
13:44:28 Primary ZLU is  801
ZTERM> rc 10 20
ZTERM> tx 10 50 0 2 2
13:44:45 TX test complete!!!
ZTERM>
```

Step 11. The sent data can now be received and displayed:

```
ZTERM> rx list
13:44:51 Msg from tmnl ZLU#00020(00123) len   50 No error detected
 Lcn .00. .01. .02. .03. .04. .05. .06. .07. .08. .09.  ASCII.....ASCII.....
 000 5A74 3031 3534 3100 206D 6573 7361 6765 206E 6F2E  Zt01541  message no.
 010 2030 3030 3031 206C 656E 6774 6820 2020 2035 3024   00001 length    50$
 020 6F70 7172 7374 7576 7758                           opqrstuvwX
13:44:51 Msg from tmnl ZLU#00020(00123) len    50 No error detected
 Lcn .00. .01. .02. .03. .04. .05. .06. .07. .08. .09.  ASCII.....ASCII.....
 000 5A74 3031 3534 3100 206D 6573 7361 6765 206E 6F2E  Zt01541  message no.
 010 2030 3030 3032 206C 656E 6774 6820 2020 2035 3024   00002 length    50$
 020 6F70 7172 7374 7576 7758                           opqrstuvwX

    <CNTL-C>

13:44:53 Error on ZREAD: Interrupt occurs while waiting
13:44:53 Messages received         2,    Messages sent            0
13:44:53 Elapsed time secs         2,    Messages/sec             1
13:44:53 Data in KBytes            0,    Data rate (KB/s)         0
13:44:54 Sys CPU   0%, User CPU   2%,    Total CPU   2%
13:44:54 Mux   0 utilization      1%
ZTERM> ex
```

Step 12. The ASCII part of the displayed message can be used to observe the message number. The *rx* command must be terminated using a <cntrl-C> command.

3 Using HDLC/LAP-B (ABM) Protocol

Introduction

For a complete description of the communication formats, refer to the standards documents mentioned at the start of this manual.

Each unit transmitted over an HDLC/LAP-B (ABM) link is a “frame”. A frame has an address field, a control field and a trailing CRC (Frame Check Sequence), is preceded and followed by a flag byte (binary 01111110), and may include an 'I' (information) field as well. Frames may be 'unnumbered' (link control) frames, flow control/supervisory frames or information frames. The control fields of flow control and information frames contain frame-level sequence numbers.

The maximum length of the (received) I-field must be established and agreed upon for any given HDLC/LAP-B (ABM) link, so that buffering requirements may be met and appropriate timer values set. The ACC HDLC/LAP-B (ABM) implementation will allow agreed maximum I-frame sizes of up to 16,383 bytes, subject to backplane limitation. See the section “Poll/Select Configuration Values” in Chapter 4 ,“Protocol Specific Configuration” for further details.

Application Message Headers

Normal HDLC/LAP-B (ABM) Mode

All application data is transmitted (or received) as the I-field of an information frame. All headers and/or trailers are supplied by the protocol for outgoing messages and removed by the protocol for incoming messages. Only the I-field data to be transferred is dealt with by the application program, which need not even be aware that HDLC/LAP-B (ABM) is being used for the communications.

Timeout Processing

Frame timeouts are set in the Poll and Select configuration words. The T1 timer defines the period within which a response to a Command frame must be received across the HDLC/LAP-B (ABM) link. Should this timer expire before the expected response is received, then the Command frame will be re-transmitted up to a pre-determined maximum number of times (N2). The maximum number of re-transmissions is also set by the Poll and Select configuration words. If the timer expires on that number of successive occasions, then the LAP-B link will be reset.

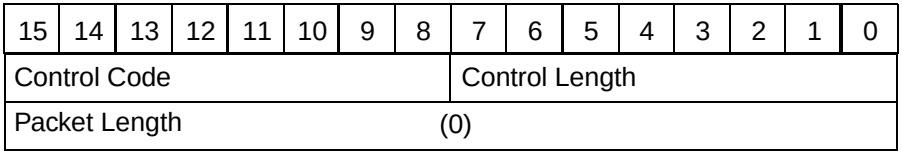
An additional timer referred to as T2 is used to control the transmission of acknowledgments to received I-frames. As the acknowledgment may be 'piggy-backed' onto a transmitted I-frame, it may not always be necessary to transmit an explicit flow control frame to send the acknowledgment. When an I-frame is received, timer T2 is started, and if an I-frame is transmitted, the acknowledgment is sent with it, and timer T2 will be cancelled. Only if T2 expires (no I-frame having been transmitted) will an explicit flow control frame be transmitted. T2 is always set to half of the value of the T1 timer.

Request Specific Processing

Control Requests (E1/T1 card only)

All control writes to this protocol share a common format as shown in Figure 3-1. Control writes are issued to the card using the *zcntl()* API call.

Figure 3-1 Control Request Format



The first four bytes are defined as a header:

Control Code — Defines the action to be taken by this request.

Control Length — The length of the data for this control request. This data immediately follows the header.

Packet Length — Some protocols allow packet data (data to be transmitted over the wire) to be passed to a control write following the control request data. For this protocol, these two bytes must be zero because the protocol does not allow packet data within a control write.

Control Codes

This protocol allows the following Control Codes:

5 CW_STATS Send ST25L2STAT status message

This control write triggers the protocol to send a ST25L2STAT status message. For the format of this message, see the section on “Status and Error Messages”. Control length must be zero.

6 CW_REVCODE Send ST25REVCD status message

This control write triggers the protocol to send a ST25REVCD status message. For the format of this message, see the section on “Status and Error Messages”. Control length must be zero.

7 CW_TIMERS Set timers

The protocol’s timers may be changed from their configured values using this control write. This request should be used with caution because the values configured with this request will not be restored if the card is reset by ZCOM.

The control request data consists of three pairs of bytes. Each pair of bytes consists of a Resolution and a Value. The legal values for Resolution are as follows:

0 = don’t set
1 = 10msec
2 = 100msec
3 = 1sec
4 = 10sec.

Note that these values are different from the values used for the resolutions when configuring a terminal. The Values are used directly. A zero Value will disable the timer if the resolution is non-zero.

The first pair of bytes applies to T1, the second pair is ignored (should be set to zero) and the third pair apply to T3/T4. See Figure 3-2 on page 35.

Figure 3-2 **Protocol Timers**

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T1 Resolution (0-4)								T1 Value (0-255)							
(0)															
T3/T4 Resolution (0-4)								T3/T4 Value (0-255)							

Status and Error Messages

HDLC/LAP-B (ABM) makes use of a range of message status codes and unsolicited status messages, other than the standard status codes described in the *ACC Error Guide*.

The definitions for these status codes are located in /usr/include/zcomstatus.h and /usr/include/zcom/zx25status.h.

Terminal Request Status Codes

The following statuses are returned in response to a request to create, delete, enable, disable, activate or deactivate a terminal. Although these are a part of the standard status set, they are not used by all protocols, and have a particular significance within HDLC/LAP-B (ABM).

0 TM_OK No error detected

The request was successfully processed. In the case of enabling or disabling the terminal, this does not mean that the link has been established or disconnected. The link status is reported via unsolicited status messages.

1 TM_MUST_DSBL Terminal must be disabled

This status code is returned in response to a terminal delete request. Before deleting a terminal, it must be disabled and the ST25DSBL unsolicited status message must have been received.

12 TM_PROT_REJ Protocol detected error

TM_PROT_REJ is returned in response to a terminal enable request when the configuration specified by the poll, select and option words is invalid. In addition to returning TM_PROT_REJ, the protocol generates a ST25ENF unsolicited status message. For configuration information, see Chapter 4 , “Protocol Specific Configuration.”

14 TM_TABLE_SIZE Protocol table sizes too small

The configured terminal table size is too small to allow LAP-B to run. This status is returned in response to a terminal creation request.

Unsolicited Status Messages

This section describes the unsolicited status messages which may be received on the ZLU which is set as the receiver for a terminal using this protocol. The status codes received may or may not have bit 7 set. Bit 7 is the down bit (masked by IO_DOWN_MASK). The down bit is used as an indicator of the link status. When an error occurs on the link or the link is enabled after being disabled, the protocol attempts to enter multiple frame acknowledged operation. Until the attempt succeeds or fails, the down bit is clear (i.e. the terminal is considered to be up until proven otherwise). If the attempt subsequently fails, the down bit is set; otherwise the down bit is left cleared.

The following status messages are generated in response to terminal enable and disable requests:

0 IO_OK No error detected

This status message is generated in response to a terminal enable request after multiple frame acknowledged operation is established. Note that the down bit will always be clear (bit 7 = 0).

41 IO_ALRDY_ENBL Terminal already enabled

This status message is generated in response to a terminal enable request when the terminal is already enabled and the protocol is not currently trying to establish multiple frame acknowledged operation. If the protocol is currently trying to establish multiple frame acknowledged operation, this status message will not be sent. Instead, either a ST25ENBL with the down bit set or an IO_OK status message will be sent.

42 IO_ALRDY_DSBL Terminal already disabled

This status message is generated in response to a terminal disable request when the terminal is already disabled. The down bit will always be set.

64 ST25ENBL Link NOT established on ENABLE

This status message is generated in response to a terminal enable request after it is determined that multiple frame acknowledged operation can not be established. The down bit will always be set.

65 ST25DSBL Link disconnected on DISABLE

This status message is generated in response to a terminal disable request after multiple frame acknowledged operation has terminated. The down bit will always be set.

77 ST25ENF ENABLE failed - bad config

This status message is generated in response to a terminal enable request when the terminal's configuration is invalid. The down bit will always be set.

The following unsolicited status messages are sent after an error occurs when the link is in multiple frame acknowledged operation. The error conditions include cable faults, inability to receive acknowledgment of transmitted frames or receiving frames that are inconsistent with multiple frame acknowledged operation. When one of these error conditions occurs, the protocol attempts to re-establish multiple frame acknowledged operation. After the attempt has either succeeded or failed, the unsolicited status message that reflects the original error condition is sent. If the attempt fails, the link is considered disconnected and the down bit will be set in the status; otherwise the link has only been reset and the down bit will be clear.

66 ST25XDCD Link disc. on loss of carrier

The card is no longer receiving the required communications signals. This can be caused by incorrect port configuration or disconnected/faulty cabling.

68 ST25RTRY Exceeded retransmit limit

A frame was transmitted N2 times without receiving an acknowledgment from the remote device.

69 ST25TXFR Reset/Disc. due to transmitted FRMR

A frame containing an error was received by the protocol; this causes LAP-B to reset the link by sending a FRMR. For consistent error handling between LAP-B and LAP-D (which sends a SABME instead of a FRMR), the latter protocol sends this status message. (The data buffer contains the information field of the FRMR that LAP-B would send.)

70 ST25RXFR Reset/Disc. due to received FRMR

A frame reject response was received. A copy of the received FRMR's information field is in the data buffer of the unsolicited status message.

71 ST25RXDM Reset/Disc. due to received DM

The protocol received a Disconnected Mode response.

72 ST25RXSA Reset/Disc. due to recvd SABM/SABME

A Set Asynchronous Balanced Mode (or SABME) command was received by the protocol.

73 ST25RXDI Reset/Disc. due to received DISC

The protocol received a DISConnect command.

74 ST25RXUA Reset/Disc. due to received UA

An Unnumbered Acknowledgment response was received by the protocol when the protocol had not solicited a UA.

75 ST25RUFR Reset/Disc. due to unsol final resp

The protocol received a response with the Final bit set which had not been solicited by the transmission of a command with the Poll bit set.

The following status messages are used by the protocol on E1/T1 hardware to return data in response to a control write:

88 ST25L2STAT Statistics upload

This message is sent in response to a CW_STATS control write. The data buffer contains a x25l2stat_type structure (defined in /usr/include/zcom/zcomx25.h).

89 ST25REVCD Revision code upload

This message is sent in response to a CW_REVCODE control write. The data buffer contains a single byte which is the current protocol version. Currently, this is 11.

Write Completion Status Codes

The following statuses are returned in response to write requests. Although these are a part of the standard status set, they are not used by all protocols, and have a particular significance within this protocol.

0 IO_OK No error detected

The request was successfully transmitted and acknowledged by the remote side.

1 IO_DSBL Terminal disabled

The terminal must be enabled before it will accept write or control write requests.

22 IO_LNK_DSC Link disconnected

An unrecoverable error occurred while this request was waiting to be sent or to be acknowledged by the remote side. The transmit request has been flushed.

23 IO_LNK_RST Link reset

A recoverable error occurred while this request was waiting to be acknowledged by the remote side. The transmit request has been flushed.

Control Write Completion Status Codes

The following statuses are returned by the LAP-B protocol on E1/T1 hardware in response to control write requests. Although these are a part of the standard status set, they are not used by all protocols, and have a particular significance within this protocol.

0 IO_OK — No error detected

The request was successfully transmitted and acknowledged by the remote side.

1 IO_DSBL — Terminal disabled

The terminal must be enabled before it will accept write or control write requests.

11 IO_LONG_MSG — Message too long

Indicates one of the following errors:

- The packet length in the header is not zero.
- The control length in the header is not zero for a CW_STATS or CW_REVCODE request.
- The control length in the header is greater than six for a CW_TIMERS request.

12 IO_SHRT_MSG — Message too short

Indicates one of the following errors:

- The request length is less than four (too short to contain the control write header).
- The control length in the header is less than six for a CW_TIMERS request.

24 IO_BAD_CTL — Bad control function or format

Indicates one of the following errors:

- The control length in the header does not equal the request length plus four.
- The control code in the header contains a value other than CW_STATS, CW_REVCODE or CW_TIMERS.
- The value specified for a timer resolution in a CW_TIMERS request is outside of the range zero to four.

Read Completion Status Codes

All receive completion messages will have the status IO_OK. Received messages that have errors are not returned by the protocol.

4 Protocol Specific Configuration

Introduction

This section provides specific information on preparing the network configuration file when HDLC/LAP-B (ABM) is to be used.The parts of the network configuration (`.answ`) file relevant to HDLC/LAP-B (ABM) are:

- Interface-definition
- Port-definition
- Terminal-definition

Interface-Definition

An Interface-Definition line appears in the network configuration file for each ACC card that will have LAP-B (ABM) connections configured on it.

It is possible to use other protocols with HDLC/LAP-B (ABM) on a single card, provided the firmware file has been customized to include those other protocols. (Refer to the sections on zlink in the *ACC Installation and Configuration Guide* and the *ACC Utilities Reference Guide* for more information on creating firmware files for multiple protocols.)

Sample Interface-Definition statements are as follows:

```
Z7200A     0      0:4/opt/acc/z7200a/hdlc.zabs/* 8-port card */
Z7350A     1      0:8/opt/acc/z7350a/hdlc.zabs/* 2-port card */
Z7300A     2      0:16/opt/acc/z7300a/hdlc.zabs/* 4-port E1/T1 */
```

Port-Definition

On 2-port or 8-port cards, the port must be defined as operating in SDLC mode. The ACC cards may use either an external (modem supplied) clock, or an internal (card supplied) clock. With an external (modem supplied) clock, the speed is for documentation purposes only, and will not be used by the ZCOM system.

The clock multiplier should be x1, and the encoding mode would normally be NRZ (although NRZI may be used).

A sample Port-Definition is as follows:

```
Port 01:04 RS232 57600   Ext    SDLC  x1     NRZ
```

For the E1/T1 ACC card there are no Port-Definition requirements specific to the HDLC/LAP-B (ABM) protocol. A typical E1/T1 Port-Definition would be:

```
Port  0:2  E1RJ45  Ext  CRCMF  HDB3
```

Refer to the section on the TTGEN utility in the *ACC Utilities Reference Guide* for more information on specifying the port and subchannel parameters.

Terminal-Definition

Each HDLC/LAP-B (ABM) port or E1/T1 subchannel requires one (and only one) Terminal-Definition for the HDLC/LAP-B (ABM) point-to-point link. The LAP-B terminal is defined by a TERM line in the LAP-B sample file (`lapb_sample.answ`).

Some sample HDLC/LAP-B (ABM) Terminal-Definition lines follow:

```
Term   120   1:6 HDLC.LAPB   HDLC_DTE_400MS     1000 0 0 0 0 ″HDLC link 1″
```

The example above is a Terminal-Definition that defines an HDLC/LAP-B (ABM) terminal with ZLU 120 on ACC card 1 port 6. It is configured as a DTE with frame time-out value of 400 milliseconds.

```
Term   120   1:6 HDLC.LAPB   8080H 146AH        1000 0 0 0 0 “HDLC link 1″
```

The second definition is the same as the first except that it uses the POLL and SELECT words to define the mode of operation. When the POLL and SELECT values are used they are specified as two 16 bit hex values.

The descriptions of the fields within these values are described in the section “Poll/Select Configuration Values” in this chapter.

The Terminal-Definition takes the following format:

```
Term   zlu   card:port:subc   type   poll   select   application_data   name <option>
```

The following list gives more specific information on the parameters in the HDLC/LAP-B (ABM) Terminal-Definition.

ZLU — This number (120 in the example) assigns a unique reference number for this terminal.

Card:port:subc — The card number must have been defined in an Interface-Definition statement. The port must have been defined in a Port-Definition statement. The “:subc” information must be supplied for a 4-port E1/T1 card and should be omitted for all other card types. The subchannel so specified must have been defined in a Subchannel- statement.

Device — The device type of the HDLC/LAP-B (ABM) terminal must be HDLC.LAPB.

Config	This configuration keyword supplies the 'POLL' and 'SELECT' parameters for the ZCOM terminal definition. These are used to specify: • whether the HDLC/LAP-B (ABM) terminal acts as a DTE or DCE, • the frame re-transmission time-out value, • the re-transmission count limit, • the outstanding frame window size to be used, • the maximum frame size allowed. In most cases a pre-configured keyword will suffice. The currently defined keywords are given in the subsection "Predefined Configuration Values" of this section. If a different set of values is required, then different values for the 'POLL' and 'SELECT' codes may be entered here in the standard poll and select format in hexadecimal.
appl. no.	Application number (Application dependent - has no effect on protocol).
inst. no.	Institution number (Application dependent - has no effect on protocol).
brch. no.	Branch number (Application dependent - has no effect on protocol).
wkst. no.	Workstation number (Application dependent - has no effect on protocol).
area no.	Area number (Application dependent - has no effect on protocol).

NOTE The five parameters above are for the use of the application and can be set to any value desired. They can be accessed by the application from the Logical Terminal Table using the zinfo() library function.

name	Choose a meaningful description of the LAP-B link, preferably including a reference to the location of the remote end of the link. This field will be used in some ZMNTR displays. It may also be acessed programmatically using the zinfo() library function.

Predefined Configuration Values

The following configuration values are currently defined for use in HDLC/LAP-B (ABM) configuration as a replacement for the POLL and SELECT address fields. These values are used (for example HDLC_DTE_3.0) as a symbolic way of configuring the associated two 16-bit hexadecimal values.

If a particular desired configuration value is not provided by these values, then two hexadecimal values can be used instead (as for normal poll and select specification).

```
HDLC_DTE_400MS   and   HDLC_DCE_400MS

    I-field = 0128 bytes
         T1 = 400ms    T2 = 200ms    (10ms * 20)
          K = 3        N2 = 10

    Equivalent parameters
     HDLC_DTE_400MS - 8080h 146Ah
     HDLC_DCE_400MS - 0080h 146Ah

HDLC_DTE_3.0   and   HDLC_DCE_3.0

    I-field = 1024 bytes
         T1 = 3.0sec    T2 = 1.5sec (100ms * 15)
          K = 3,        N2 = 10

    Equivalent parameters
     HDLC_DTE_3.0   - 8400h 4F6Ah
     HDLC_DCE_3.0   - 0400h 4F6Ah
```

CAUTION DO NOT make changes to the `/opt/acc/cfg/ttgendefine` file itself. This file is part of the ACC Base product software, and may be overwritten when updates to the product are applied.

NOTE Specific user defined configurations may be added to the network configuration file. These configuration should use the `#define` statement and be placed at the beginning of the file. Refer to the examples in the file `/opt/acc/cfg/ttgendefine`.

Poll/Select Configuration Values

Poll Word

DTE/DCE	Operating mode - DTE or DCE
Max I-field	Largest I-field allowed (in bytes)

Select Word

T1	L2 acknowledgment Timer
K	Frame level Tx window
N2	Number of Tx retries allowed

Poll Word

Figure 4-1 Poll Word Format

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T/C	-0-	Maximum I-Field size (1-16383 bytes) See following note.													

T/C	Configure port as a DCE or DTE: 0 = Port to act as DCE 1 = Port to act as DTE
Max I-field	Maximum number of bytes is the largest acceptable I field. Values from 1 to 10,066 bytes. Note that the backplane mechanism limits the largest possible transmitted size to 10,066 bytes (rather than 16,383).

NOTE Some ACC interface cards have limited buffering capacity. Large window sizes should be used with caution, especially with large frame sizes, as the buffering capacity of the ACC interface may be exceeded.

Select Word

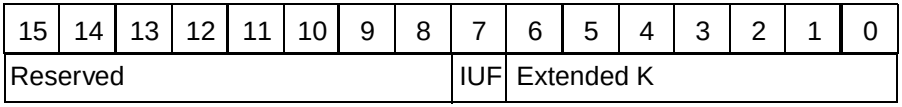
Figure 4-2 Select Word Format

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T1 Res		T1÷2 (1-63)						K (0-7)			N2 (1-31)				

T1÷2	This is the value of the T1 timer (divided by two). The valid range is 1 to 63; 0 is not allowed and is an error. The units are given by the T1 Res (resolution) field (see below).
T1 Res	Resolution of the T1 value above. The values correspond as follows: 0=10ms, 1 = 100ms, 2 = 1s, 3 = 10s
K	Level 2 window size, values 1 to 7. If K=0, then extended sequence numbering will be used, and the window size is configured via the “Extended K” field. If K=0, the link is initialized via a SABME command rather than the SABM command, used for standard sequence numbering.
N2	Number of retransmissions, values 1 to 31. Note that 0 is not allowed and is an error.

Option Word

Figure 4-3 Option Word Format



IUF — Error handling, normal when 0. This means normal error handling (level 2 link reset) on receipt of unsolicited response frame with F=1. 1 means Ignore F=1 (treat as F=0) in any unsolicited response frames.

Extended K — Window size for extended sequence numbering. When K=0, extended sequence numbering is selected. The Extended K field is used to configure the windows size for modulus 128 sequence numbering. Valid values for this field are from 1 to 127 inclusive.

A Sample Configuration Files

Sample Network Configuration (.answ) file

The file shown below represents the sample network configuration (.answ) file that is provided with the HDLC/LAP-B (ABM) protocol product.

```
#include "/opt/acc/cfg/ttgendefine"
ttgen
   Configuration
      System-name     "ACC HDLC/LAP-B protocol ttgen input file"
      Program-zlu     100
      Terminal-zlu    800
      Logical-term    800
      Physical-term   800
      Buffer-pool     3000000
      Logical-size    252
      Queue-Limit     200
      Transmit-limit  100
      Unack-limit     5000
      Port-limit      20000
      E1T1-Port-limit 100000
      Node-entry      1


Interface-Definition
      z7200a  0       0:4   /opt/acc/z7200a/hdlc.zabs
      z7300a  1       0:8   /opt/acc/z7300a/hdlc.zabs
*/ The "z7200a" and "z7300a" parts will be different depending
*/ on the ACC card used.

Port-Definition
      Port 0:0 RS232 57600 Int SDLC X1 NRZ
      Port 0:1 RS232 57600 Ext SDLC X1 NRZ
      Port 0:2 RS232 57600 Int SDLC X1 NRZ
      Port 0:3 RS232 57600 Ext SDLC X1 NRZ
      Port 0:4 RS232 57600 Int SDLC X1 NRZ
      Port 0:5 RS232 57600 Ext SDLC X1 NRZ
      Port 0:6 RS232 57600 Int SDLC X1 NRZ
      Port 0:7 RS232 57600 Ext SDLC X1 NRZ

Port 0:0  E1RJ45   Int  CRCMF  HDB3
      Port 0:1  E1RJ45   Ext  CRCMF  HDB3
      Port 0:2  E1RJ45   Int  CRCMF  HDB3
      Port 0:3  E1RJ45   Ext  CRCMF  HDB3
```

```
Subchannel-Definition
*     Mux 1, port 0, subchan 01, timeslot 1
      Subch 1:0:01 HDLC INV=off ITBS=2 ~
         ts:01:FFh

*     Mux 1, port 0, subchan 02, timeslot 2 to 7
      Subch 1:0:02 HDLC INV=off ITBS=12 ~
         ts:02:FFh ts:03:FFh ts:04:FFh ts:05:FFh ts:06:FFh ts:07:FFh

*     Mux 1, port 0, subchan 03, timeslot 8 to 31
      Subch 1:0:03 HDLC INV=off ITBS=48 ~
         ts:08:FFh ts:09:FFh ts:10:FFh ts:11:FFh ts:12:FFh ~
         ts:13:FFh ts:14:FFh ts:15:FFh ts:16:FFh ts:17:FFh ~
         ts:18:FFh ts:19:FFh ts:20:FFh ts:21:FFh ts:22:FFh ~
         ts:23:FFh ts:24:FFh ts:25:FFh ts:26:FFh ts:27:FFh ~
         ts:28:FFh ts:29:FFh ts:30:FFh ts:31:FFh

*     Mux 1, port 1, subchan 01, timeslot 1
      Subch 1:1:01 HDLC INV=off ITBS=2 ~
         ts:01:FFh

*     Mux 1, port 1, subchan 02, timeslot 2 to 7
      Subch 1:1:02 HDLC INV=off ITBS=12 ~
         ts:02:FFh ts:03:FFh ts:04:FFh ts:05:FFh ts:06:FFh ts:07:FFh

*     Mux 1, port 1, subchan 03, timeslot 8 to 31
      Subch 1:1:03 HDLC INV=off ITBS=48 ~
         ts:08:FFh ts:09:FFh ts:10:FFh ts:11:FFh ts:12:FFh ~
         ts:13:FFh ts:14:FFh ts:15:FFh ts:16:FFh ts:17:FFh ~
         ts:18:FFh ts:19:FFh ts:20:FFh ts:21:FFh ts:22:FFh ~
         ts:23:FFh ts:24:FFh ts:25:FFh ts:26:FFh ts:27:FFh ~
         ts:28:FFh ts:29:FFh ts:30:FFh ts:31:FFh

*     Mux 1, port 2, subchan 01, timeslot 1
      Subch 1:2:01 HDLC INV=off ITBS=2 ~
         ts:01:FFh

*     Mux 1, port 2, subchan 02, timeslot 2 to 7
      Subch 1:2:02 HDLC INV=off ITBS=12 ~
         ts:02:FFh ts:03:FFh ts:04:FFh ts:05:FFh ts:06:FFh ts:07:FFh

*     Mux 1, port 2, subchan 03, timeslot 8 to 31
      Subch 1:2:03 HDLC INV=off ITBS=48 ~
         ts:08:FFh ts:09:FFh ts:10:FFh ts:11:FFh ts:12:FFh ~
         ts:13:FFh ts:14:FFh ts:15:FFh ts:16:FFh ts:17:FFh ~
         ts:18:FFh ts:19:FFh ts:20:FFh ts:21:FFh ts:22:FFh ~
         ts:23:FFh ts:24:FFh ts:25:FFh ts:26:FFh ts:27:FFh ~
         ts:28:FFh ts:29:FFh ts:30:FFh ts:31:FFh
```

```
*     Mux 1, port 3, subchan 01, timeslot 1
      Subch 1:3:01 HDLC INV=off ITBS=2 ~
         ts:01:FFh

*     Mux 1, port 3, subchan 02, timeslot 2 to 7
      Subch 1:3:02 HDLC INV=off ITBS=12 ~
         ts:02:FFh ts:03:FFh ts:04:FFh ts:05:FFh ts:06:FFh ts:07:FFh

*     Mux 1, port 3, subchan 03, timeslot 8 to 31
      Subch 1:3:03 HDLC INV=off ITBS=48 ~
         ts:08:FFh ts:09:FFh ts:10:FFh ts:11:FFh ts:12:FFh ~
         ts:13:FFh ts:14:FFh ts:15:FFh ts:16:FFh ts:17:FFh ~
         ts:18:FFh ts:19:FFh ts:20:FFh ts:21:FFh ts:22:FFh ~
         ts:23:FFh ts:24:FFh ts:25:FFh ts:26:FFh ts:27:FFh ~
         ts:28:FFh ts:29:FFh ts:30:FFh ts:31:FFh

Terminal-Definition
      Term 10 0:0 HDLC.LAPB HDLC_DTE_400MS 1000 0 0 0 0 ”HDLC link 1”
      Term 20 0:1 HDLC.LAPB HDLC_DCE_400MS 1000 0 0 0 0 ”HDLC link 2”

Term 600 1:0:3  HDLC.LAPB  1000h 46EAh 101 0 0 0 0 "HDLC link 3"
        option 0086h
      Term 601 1:1:3  HDLC.LAPB  9000h 46EAh 101 0 0 0 0 "HDLC link 4"
        option 0086h
      Term 602 1:2:3  HDLC.LAPB  1000h 46EAh 101 0 0 0 0 "HDLC link 5"
        option 0086h
      Term 603 1:3:3  HDLC.LAPB  9000h 46EAh 101 0 0 0 0 "HDLC link 6"
        option 0086h

Node-Definition
      Local-Node     123

End$
```

Download Linkage

This file is the hdlc.zlnk file that is provided for the 8-Channel HP-PB (NIO) Multiplexer interface, and is typical for other interfaces as well. (Z7200A denotes the 8-channel HP-PB interface hardware.)

```
ma z7200a/hdlc.zmap
xr
nam HDLC Download file for ACC System
sn sys/z7200_rom.zsnp
org,vmux1org,vmux1lim
re sys/wmux1.zrel
org,vmux3org,vmux3lim
re sys/wmux3.zrel
org,progorg,proglim
re sys/wmux4.zrel
re protocol/level1.zrel
***************  Protocol Modules  ***************************
re protocol/hdlcabm.zrel
**************  End of Protocol Modules  **********************
re protocol/monitor.zrel
re protocol/testprot.zrel
org,eptable
re sys/pmenttab.zrel
org,preptbl
re sys/umuxent.zrel
ab z7200a/hdlc.zabs
end
```

If any of the other ACC multiplexers which support the LAP-B level 2 protocol are in use, a different .zlink file is provided.

When customizing download files, use whichever files have been provided with your configuration.