

HP X.25 for OpenVMS

Release Notes

July 2005

These Release Notes apply to the HP X.25 for OpenVMS, Version 2.0 product. Please read these Release Notes before attempting to install this product.

Operating System:	OpenVMS I64 Version 8.2 and 8.2-1 OpenVMS Alpha Version 8.2
DECnet-Plus Software:	HP DECnet-Plus for OpenVMS Version 8.2 and 8.2-1
Software Version:	HP X.25 for OpenVMS Version 2.0

Hewlett-Packard Company
Palo Alto, California

© Copyright 2005 Hewlett-Packard Development Company, L.P.

Confidential computer software. Valid license from HP required for possession, use, or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Intel and Itanium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group.

Printed in the US

Contents

Preface	vii
1 Installation Information	
1.1 Installation Instructions	1-1
1.2 Installation/Configuration Prerequisites	1-1
2 New Features	
2.1 Support for OpenVMS I64	2-1
3 Resolved Problems	
3.1 OpenVMS V8.2-1 on I64 platform crashes whenever the data is transferred over a XOT DTE	3-1
3.2 X.25 applications do not work using PVC and GAP	3-1
3.3 PBXDD does not assert DTR when cable is removed/reinserted	3-1
3.4 The maximum number of DTEs that can be created is 100	3-1
3.5 Deleting a DTE using the configurator does not remove all the DTE attributes	3-1
3.6 OpenVMS V8.2 crashes when configured with X.25 relay client	3-2
3.7 Reconfigured LLC2 DTE resets the values to default when the user steps through the current values	3-2
3.8 System Crash with RSVD_LP in GAP_DELETE_GSB	3-2
3.9 PBXDD Buffer Too Small for SNA	3-2
3.10 M-bit Incorrectly Handled for Server-Client Connections	3-2
3.11 Pool Corruption When Using CUG with Server-Client Connections	3-2
3.12 X.25 Relay Hangs with Depleted Credits	3-2
3.13 Rights Identifiers Corrupted by X25\$CONFIGURE	3-3
3.14 X25\$CONFIGURE Doesn't Allow PVC Configuration with XOT	3-3
3.15 X25\$SHUTDOWN.COM Doesn't Delete Some X.25 Entities	3-3
3.16 Multiple Issues with XOT	3-3
3.17 System Crash at X25\$L1+11F70	3-3
3.18 Invalid Security State Bugcheck	3-3
3.19 Machine Check Crash During Load of X25\$ZZDRIVER	3-3
3.20 Double Deallocation Bugcheck in X25\$ACCESS.EXE	3-4
3.21 VCRP Leak in Nonpaged Pool	3-4
3.22 ZZ Device Shows Offline	3-4
3.23 Inverted Clock Required with PBXDD	3-4

4 Known X.25 Problems

4.1	Several X.25 Server and X.25 Relay Characteristics Not Implemented . . .	4-1
4.2	X.29 Logins Hang and Incorrect Terminal Naming with High Device Numbers	4-1
4.3	X25\$APPLD Unexpectedly Exhausts BYTLM Quota	4-1
4.4	Facility Code "Called Address Extension" Is Modified by the X.25 Software for CALLC Packet	4-2
4.5	Error Messages During Reinstall/Upgrade of X.25	4-2

5 Documentation Errata and Addenda

5.1	Manual References	5-1
5.2	VAX WANDD Programming Manual (OpenVMS Alpha)	5-1
5.2.1	General Information	5-1
5.2.1.1	Supported Devices	5-1
5.2.1.2	Framing	5-1
5.2.1.3	Half-Duplex and Multipoint Operations	5-2
5.2.1.4	LLC2 Datalink	5-2
5.2.2	Specific Corrections to the WANDD Programming Reference Manual	5-2
5.2.2.1	Section 4.2.1 Open a Port	5-2
5.2.2.2	Section 4.2.2 Enable Attention AST	5-2
5.2.2.3	Section 4.2.5 Getting Port Information	5-2
5.2.2.4	Appendix B: User-Written Datalink Protocols	5-2
5.2.2.5	Appendix C: Example Programs	5-3
5.2.2.6	Appendix D Obsolete Features of the \$QIO Interface	5-3
5.2.2.7	Section E.2 NCL Commands	5-3

6 Installation of PBXDP and PBXDI Cards (OpenVMS Alpha)

6.1	Installation of PBXDP-Ax Cards	6-1
6.2	Installation of PBXDI-Ax Cards	6-1

Preface

HP X.25 V2.0 for OpenVMS Alpha and I64 systems replaces the previous version of X.25 software. It includes all changes and bugfixes since V1.6.

Note

X.25 for OpenVMS Version does not support any synchronous data link hardware on OpenVMS I64 systems.

Intended Audience

This manual is intended for all users of HP X.25 for OpenVMS.

Structure of This Book

This document contains the following chapters:

- Chapter 1, *Installation Information*
- Chapter 2, *New Features*
- Chapter 3, *Resolved Problems*
- Chapter 4, *Known X.25 Problems*
- Chapter 5, *Documentation Errata and Addenda*
- Chapter 6, *Installation of PBXDP and PBXDI Cards (OpenVMS Alpha)*

X.25 V2.0 for OpenVMS I64 Systems Methods for Communication

The X.25 V2.0 release for OpenVMS I64 system provides support for communication via:

- GAP (X.25 over DECnet)
- LLC2 (X.25 over LAN)
- XOT (X.25 over TCP/IP)

These methods require an external machine to act as a gateway or relay to the X.25 connection or PSDN. This machine provides the interface for the actual synchronous line and may be implemented using an OpenVMS Alpha system with a PBXDP or PBXDD synchronous communication controller, or a third party dedicated router.

X.25 native communication using a synchronous communication controllers on OpenVMS I64 system will not be supported.

For more information please refer to the *HP X.25 for OpenVMS- Configuration manual*.

Reader's Comments

HP welcomes your comments on this manual or any of the X.25 for OpenVMS documents. Please send comments to either of the following addresses:

Internet	openvmsdoc@hp.com
Mail	Hewlett-Packard Company OSSG Documentation Group, ZKO3-4/U08 110 Spit Brook Rd. Nashua, NH 03062-2698

Installation Information

1.1 Installation Instructions

If you are installing HP X.25 for OpenVMS software from the CD-ROM distribution, follow the instructions contained in the CD-ROM booklet.

Otherwise, install the product with the `PRODUCT` command by logging into the `SYSTEM` account, and typing the following at the DCL prompt:

```
PRODUCT INSTALL X25/SOURCE=[location of the kit]
```

The kit location may be a tape drive or a disk directory that contains the PCSI kit.

The system must be rebooted for the new images to be loaded.

1.2 Installation/Configuration Prerequisites

The following are the prerequisites for HP X.25 Version 2.0:

Operating System:	OpenVMS I64 Version 8.2 and 8.2-1 OpenVMS Alpha Version 8.2
DECnet-Plus Software:	HP DECnet-Plus for OpenVMS Version 8.2 and 8.2-1

This chapter describes the new features in this release.

2.1 Support for OpenVMS I64

This release adds support for OpenVMS I64 systems.

Important

X.25 for OpenVMS does not support any synchronous data link hardware on OpenVMS I64 systems.

Resolved Problems

This chapter describes problems resolved since X.25 for OpenVMS Version 1.6.

3.1 OpenVMS V8.2-1 on I64 platform crashes whenever the data is transferred over a XOT DTE

OpenVMS V8.2-1 on I64 platform crashes, whenever the data is transferred over a XOT DTE. The problem has been fixed in this release.

Source: Engineering Testing

3.2 X.25 applications do not work using PVC and GAP

The GAP server on Alpha does not handle the RESET IND packet from the GAP client properly, causing the application to abort. The GAP server code has been modified to fix this problem.

Source: QXCM1000221538

3.3 PBXDD does not assert DTR when cable is removed/reinserted

When the cable is removed/reinserted from the PBXDD, the NCL "show modem connect line" command indicates that DTR is asserted but the same is not indicated by the breakout box. The problem has been fixed in this release.

Source: QXCM1000195198 and QXCM1000201831

3.4 The maximum number of DTEs that can be created is 100

The maximum number of DTEs that can be created in a node is 100. When the 101st DTE is started, an error message saying "no resource available error" is displayed. The management module has been modified to correct this problem. Now the maximum limit is 250.

Source: QXCM1000226451

3.5 Deleting a DTE using the configurator does not remove all the DTE attributes

When a DTE is deleted, all the attributes of a DTE are deleted from the NCL script except the Local DTE Class. This causes display of error messages during startup. The problem has been fixed in this release.

Source: QXCM1000207698

3.6 OpenVMS V8.2 crashes when configured with X.25 relay client

OpenVMS V8.2 crashes when X.25 is restarted following the Relay client configuration. The relay code has been modified to prevent the problem.

Source: Engineering Testing

3.7 Reconfigured LLC2 DTE resets the values to default when the user steps through the current values

When modifying a LLC2 DTE with the X25\$CONFIGURE utility, the values of the characteristics will be reset to defaults, even when the user only steps through the current values. The problem has been fixed in this release.

Source: QXCM1000212246

3.8 System Crash with RSVD_LP in GAP_DELETE_GSB

If a network error caused DECnet to terminate a Session Connection being established by GAP, in some situations this would result in a corrupt X.25 data structure which would lead to a system crash. The GAP code has been modified to prevent this situation.

Source: CFS.97089

3.9 PBXDD Buffer Too Small for SNA

The buffer used by the PBXDD to receive inbound data did not correctly account for the size of the CRC in the HDLC header. The ZZDRIVER has been modified to correctly account for the CRC.

Source: CFS.101193

3.10 M-bit Incorrectly Handled for Server-Client Connections

The M-Bit was not correctly forwarded for Server-Client connections. The X.25 Server code has been modified to correctly forward the M-Bit.

Source: CFS.101769

3.11 Pool Corruption When Using CUG with Server-Client Connections

Non-Paged pool was inadvertently being corrupted when Closed User Groups were implemented over Server-Client connections. The code has been modified to correct this issue.

Source: CFS.102768

3.12 X.25 Relay Hangs with Depleted Credits

In situations where large application packets were sent via X.25 Relay, which resulted in multiple packets with the M-Bit being set, X.25 Relay would deplete the available credits for reception and the connection would hang. X.25 Relay has been modified to provide more credits for data reception.

Source: CFS.103086

3.13 Rights Identifiers Corrupted by X25\$CONFIGURE

When configuring X25 Security, if long names were used for rights identifiers, the identifier name could be corrupted, resulting in a failure to add the identifier. The code has been modified to correct the corruption issue.

Source: CFS.103637

3.14 X25\$CONFIGURE Doesn't Allow PVC Configuration with XOT

X25\$CONFIGURE would not allow a PVC circuit to be configured over a XOT DTE. The X.25 Configurator has been corrected so that PVC's can be created on XOT DTE's.

Source: CFS.103758

3.15 X25\$SHUTDOWN.COM Doesn't Delete Some X.25 Entities

X25\$SHUTDOWN.COM was not correctly deleting X.25 Relay or XOT entities. The command procedure has been modified to correctly delete these entities.

Source: CFS.104808

3.16 Multiple Issues with XOT

In performing interoperability testing with other vendors, multiple issues with XOT were encountered. These issues have been corrected.

Source: Engineering and Customer Testing

3.17 System Crash at X25\$L1+11F70

In certain situations, the ZZDRIVER could receive a message from the PBXDD card after the port associated with the card had been deleted. The driver has been modified to correctly discard any messages received for a nonexistent port. Additional modifications were required to fully correct the problem.

Source: CFS.100759 and 93541

3.18 Invalid Security State Bugcheck

If a program made multiple QIO/IO\$_ACPCONTROL calls to an NW device and the SYSGEN parameter SYSTEM_CHECK was set to a non-zero value, the system would crash when the I/O channel to the NW device was deassigned. The NWDRIVER has been modified to properly account for I/O operations.

Source: Engineering testing

3.19 Machine Check Crash During Load of X25\$ZZDRIVER

On ES47, ES80 or GS1280 hardware platforms, the system could crash during boot if a PBXDD card was installed on the system. The driver has been modified to increase the time between loading the PBXDD firmware and checking to see if the firmware is running, thus preventing PCI bus read timeouts.

Source: Hardware qualification testing

3.20 Double Deallocation Bugcheck in X25\$ACCESS.EXE

A change implemented in V1.6 to prevent a memory leak resulted in a double deallocation of memory under certain circumstances. The code has been modified to prevent the double deallocation.

Source: CFS.98308

3.21 VCRP Leak in Nonpaged Pool

A number of memory leaks associated with X.25 Server functionality were fixed.

Source: CFS.98567

3.22 ZZ Device Shows Offline

On some hardware platforms, the PBXDD controller was not initializing correctly resulting in the ZZ device showing as offline. The PBXDD driver has been modified to assert the board reset signal for a longer period of time to ensure that the PBXDD correctly resets at boot.

Source: CFS.98335

3.23 Inverted Clock Required with PBXDD

The PBXDD card required that the clock signal on V.11 cables be inverted with respect to the requirements that the PBXDP card has. The firmware for the PBXDD card was modified so that the clock does not need to be inverted.

Source: CFS.96901

Known X.25 Problems

This chapter describes the known problems with X.25 for OpenVMS V8.2.

4.1 Several X.25 Server and X.25 Relay Characteristics Not Implemented

Several characteristics of the X.25 Server and X.25 Relay entities appear to be implemented but are not actually used by the entity. These characteristics include the following:

- X25 Server module
 - Default Rights Identifiers
 - Incoming Session Template
- X25 Relay module
 - Client entity
 - + Rights Identifiers

4.2 X.29 Logins Hang and Incorrect Terminal Naming with High Device Numbers

A new feature was introduced with OpenVMS V7.3-2 that allows device numbers to exceed 9999. When the NV device number increments past 9999, `logout.exe` fails and X.29 login attempts appear to hang. In addition, `SHOW USER`, `SHOW PROCESS` and `$GETJPI` with `JPI$_TERMINAL` return truncated information about the NV terminal name which may result in display anomalies or unexpected behavior for programs that use this information.

Both problems can be eliminated by setting bit 2 in `SYSGEN`'s `DEVICE_NAMING` parameter, on most systems this results in `DEVICE_NAMING` having a value of 4. Setting bit 2 causes OpenVMS to use the previous behavior of wrapping device numbers back to zero once they exceed 9999.

Source: Engineering Testing

4.3 X25\$APPLD Unexpectedly Exhausts BYTLM Quota

This problem was seen when a customer migrated an X.29 application from the VAX X.25 implementation (VAX P.S.I.). The VAX X.25 systems were used to accommodate hundreds of simultaneous server-client connections for a real-time stock exchange application provided to remote interactive X.29 users. When this application was migrated from the VAX X.25 systems, remote users began to experience a lack of connectivity to their application.

Investigation determined that this problem was the result of the `X25$APPLD` process on the X.25 for OpenVMS system quickly exhausting its BYTLM quota.

Workaround: Dramatically increase the process quota and reboot the system in order to restore the expected functionality for this application.

Source: QAR.00085

4.4 Facility Code "Called Address Extension" Is Modified by the X.25 Software for CALLC Packet

A valid received X.25 call including the called and calling address extension results in a modified called address extension in CALLC packet.

If you using the standard X.25 examples to run a send/receive via X.25 you can reproduce the problem. The application which receives the call and an NCB with called/calling address extension does an IO\$_ACCESS without modifying the content of the NCB, but the X.25 software does modify the contents.

The expected facility called address extension is modified as in this example:

```
received data value: C9 09 10 36 26 24 58 90 09 31 03
incorrectly modified value: C9 0A 12 36 26 24 58 90 09 31 03 0E
```

Workaround: None, at this time.

Source: QAR.00089

4.5 Error Messages During Reinstall/Upgrade of X.25

If X.25 is installed on a system which already has an earlier version of the product installed, the following error messages are seen:

```
%SYSTEM-W-NOSUCHFILE, no such file
.
.
.
(SYSTEM-W-NOSUCHFILE repeated a number of times)
.
.
.
%PCSIEERRCREACCT, error creating account X25$MAIL
-SYSTEM-F-DUPIDENT, duplicate identifier
%PCSIE-OPFAILED, operation failed
```

These error messages can be ignored. In order for the installation to succeed, you must answer NO to the following question:

```
Terminating is strongly recommended.
Do you want to terminate? [YES] [NO]
```

Documentation Errata and Addenda

5.1 Manual References

With the exception of the *DECnet/OSI for OpenVMS VAX WANDD Programming* and *DECnet/OSI for OpenVMS CTF Use* manuals, for any references to DECnet-Plus manuals, please refer to the Version 8.2 or Version 7.1 documents. In some cases, the names of the manuals have changed.

5.2 VAX WANDD Programming Manual (OpenVMS Alpha)

HP X.25 for OpenVMS software on OpenVMS Alpha systems now supports the WANDD programming interface previously available only for OpenVMS VAX systems.

Information on WANDD programming is contained in the *DECnet/OSI for VMS VAX WANDD Programming* manual. Exceptions for X.25 for OpenVMS software on OpenVMS Alpha systems are documented in the sections that follow.

5.2.1 General Information

5.2.1.1 Supported Devices

The only devices supported are those currently supported by the X.25 for OpenVMS software on OpenVMS Alpha systems:

- PBXDD
- PBXDP
- PBXDI
- DNSES
- DSYT1
- SCC

Ignore all devices listed in Section 1.2 and elsewhere in the *DECnet/OSI for VMS VAX WANDD Programming* manual; these devices are only supported on OpenVMS VAX systems.

5.2.1.2 Framing

BISYNC framing is now supported when using the PBXDP card.

Except as noted in the preceding paragraph, BISYNC framing, DDCMP framing, and GENBYTE framing are not supported on OpenVMS Alpha systems.

5.2.1.3 Half-Duplex and Multipoint Operations

Half-duplex is now supported when using the PBXDP card.

Except as noted in the preceding paragraph, half-duplex operations and multipoint operations are not supported on OpenVMS Alpha systems.

5.2.1.4 LLC2 Datalink

In addition to support for the HDLC and LAPB datalink protocols discussed in the *DECnet/OSI for VMS VAX WANDD Programming* manual, X.25 for OpenVMS software on OpenVMS Alpha systems supports LLC2 data links. To open a data link port to an LLC2 data link, the name of the data link specified through the P1 parameter of the IO\$_CREATE call (see Section 4.2.1 of the manual) must be of the form LLC2.sap-name.link-name. For example, to use an LLC2 link called MY-LINK which is a subentity of MY-SAP, specify a datalink name of LLC2.MY-SAP.MY-LINK.

The data link name specified in an IO\$_CREATE call (see Section 4.2.1 of the manual) can be a system logical name which translates to a string of the required form.

5.2.2 Specific Corrections to the WANDD Programming Reference Manual

The following sections provide specific information for OpenVMS Alpha systems that differs from the OpenVMS VAX information in the *DECnet/OSI for VMS VAX WANDD Programming* manual.

5.2.2.1 Section 4.2.1 Open a Port

The P1 argument to the IO\$CREATE call can also be LLC2, as described in Section 5.2.1.4 of these release notes.

The following items codes listed in Table 4-4 are ignored on OpenVMS Alpha systems:

- dll\$k_auto_start
- dll\$k_auto_restart
- dll\$k_buffer_limit

The item code dll\$k_profile listed in Table 4-4 fails, if the profile does not match.

5.2.2.2 Section 4.2.2 Enable Attention AST

The following two status codes listed in Table 4-5 are never generated on OpenVMS Alpha systems:

- dll\$m_sts_physical_loopback
- dll\$m_err_threshold_exceeded

5.2.2.3 Section 4.2.5 Getting Port Information

In Table 4-7, the dll\$k_buffer_limit item code always returns TRUE on OpenVMS Alpha systems.

5.2.2.4 Appendix B: User-Written Datalink Protocols

Only Section B.2 on HDLC Framing is applicable to OpenVMS Alpha systems. The remaining sections of Appendix B are for OpenVMS VAX systems only.

5.2.2.5 Appendix C: Example Programs

To compile the example programs in Section C.1, use the following:

```
$ cc WANDRIVER_READ_EXAMPLE.C+sys$library:sys$lib_c/lib
```

The programs in Section C.2 are supported on OpenVMS VAX systems only.

5.2.2.6 Appendix D Obsolete Features of the \$QIO Interface

The interface is supported on OpenVMS VAX systems only and is not supported on OpenVMS Alpha systems.

5.2.2.7 Section E.2 NCL Commands

The section on Issuing NCL Commands for DDCMP applies to OpenVMS VAX systems only and is not supported on OpenVMS Alpha systems.

Following is new information on issuing NCL commands for LLC2:

Issuing NCL Commands for LLC2

If you are going to use the LLC2 module you must CREATE, SET and ENABLE an LLC2 sap and link.

```
NCL> CREATE LLC2 SAP sap-name
NCL> SET LLC2 SAP sap-name LAN STATION lan-name, -
_NCL> LOCAL LSAP ADDRESS sap-addr
NCL> CREATE LLC2 SAP sap-name LINK link-name
NCL> SET LLC2 SAP sap-name LINK link-name -
_NCL> REMOTE MAC ADDRESS mac-addr
NCL> SET LLC2 SAP sap-name LINK link-name -
_NCL> REMOTE LSAP ADDRESS rsap-addr
NCL> ENABLE LLC2 SAP sap-name
NCL> ENABLE LLC2 SAP sap-name LINK link-name
```

Replace the variables like this:

<i>sap-name</i>	is the name you want to give to the LLC2 sap
<i>lan-name</i>	is the name of the LAN device over which LLC2 is to run, for example, CSMACD STATION csmacd-0
<i>lsap-addr</i>	is the value of the Local Service Access Point address, which should normally be 7E but if another value must be used an even value must be specified, for example, 7C (but not 7D or 7F)
<i>link-name</i>	is the name you want to give to the LLC2 link (note there can be multiple LLC2 links to an individual LLC2 sap)
<i>mac-addr</i>	is the 6-byte LAN MAC address of the interface to be used on the remote system, for example, AA-00-04-00-16-10
<i>rsap-addr</i>	is the value of the Remote Service Access Point address, which should normally be 7E but if another value must be used an even value must be specified, for example, 7C (but not 7D or 7F)

Other LLC2 attribute values are described in the *HP DECnet-Plus for OpenVMS—Network Control Language Reference* manual. See the *HP X.25 for OpenVMS—Configuration* manual for more information on LLC2 configuration.

Installation of PBXDP and PBXDI Cards (OpenVMS Alpha)

6.1 Installation of PBXDP–Ax Cards

PBXDP-Ax (PCI synchronous communications) cards may be simply plugged into an unoccupied PCI slot in the system. Any PBXDP–Ax cards will be automatically recognized when the X.25 or WAN software is configured.

6.2 Installation of PBXDI–Ax Cards

Installing the PBXDI–Ax card requires the following steps:

1. Invoke the console utility ICONFIG to specify the adapter configuration parameters: I/O register base, IRQ and memory base.
2. Set the adapter I/O register base through the adapter switches. The other adapter configuration parameters are set by operating system software from the values specified to the ICONFIG utility.
3. Set the jumpers for the electrical interface to be used (necessary only on boards supporting the EIA–232 and V.35 electrical interface).
4. Physically install the adapter into the system.

It is recommended that the console configuration commands (step 2) be issued **before** setting the adapter IO register base so that any console detected resource conflict can be rectified BEFORE physically configuring the adapter IO register base.

The following list describes the steps in more detail:

1. Invoke the console utility ICONFIG, for example:

```
>>> iconfig
ISA CONFIGURATION UTILITY

Name           I/O  IRQ  DMA  Mem      Length  Enabled
MOUSE          60   12
KBD            60    1
COM1           3f8   4
COM2           2f8   3
LPT1           3bc   7
FLOPPY         3f0   6    2
```

Enter PBXDI-A2 (2–port PBXDI–Ax card), PBXDI–A4 (4–port PBXDI–Ax card), PBXDI–A (generic PBXDI–Ax card, number of ports unspecified), or PBXDI (generic PBXDI–Ax card, number of ports unspecified).

The I/O base address must be a multiple of 0x10. The factory configured address is 0x300 which is not in use by any other device on this particular system so that may be selected.

The IRQ must be one of 3, 5, 7, 10, 11, 12 or 15. The selected IRQ must not be in use by any other device. On this particular system IRQ 5 is the lowest PBXDI-Ax IRQ not in use by any other device so specify that.

The PBXDI-Ax does not use a DMA channel so TAB over that field.

The Mem value must be a multiple of 0x4000 (16K). On this particular system no other device uses any ISA memory so 0 may be selected.

The Length field must be specified as 4000.

The Enabled field must contain Y otherwise the console will not tell the operating system about the device.

After configuration:

ISA CONFIGURATION UTILITY

Name	I/O	IRQ	DMA	Mem	Length	Enabled
MOUSE	60	12				Y
KBD	60	1				Y
COM1	3f8	4				Y
COM2	2f8	3				Y
LPT1	3bc	7				Y
FLOPPY	3f0	6	2			Y
PBXDI-A	300	5		0	4000	Y

Subsequent PBXDI-Ax cards could be configured with I/O 310, 320 (etc), IRQ 10, 11 (etc.) and memory 4000, 8000 (etc).

2. Set the Adapter I/O register base address. See Section 5.11 of the accompanying manual.
3. Set the appropriate electrical interface. This is required only if a board supporting V.35 or RS-232 is to be installed. See Appendix B of the accompanying manual.
4. Physically install the adapter into the system. See the appropriate system manual for details on installing expansion adapters.