endace
accelerated

DAG 3.6EP/T Card User Guide

EDM01-05

www.endace.com

Published by:

Endace Measurement Systems® Ltd
Building 7
17 Lambie Drive
PO Box 76802
Manukau City 1702
New Zealand
Phone: +64 9 262 7260
Fax: +64 9 262 7261
support@endace.com
www.endace.com

International Locations

New Zealand
Endace Technology® Ltd
Level 9
85 Alexandra Street
PO Box 19246
Hamilton 2001
New Zealand
Phone: +64 7 839 0540
Fax: +64 7 839 0543
support@endace.com
www.endace.com

Americas
Endace USA® Ltd
Suite 220
11495 Sunset Hill Road
Reston
Virginia 20190
United States of America
Phone: ++1 703 382 0155
Fax: ++1 703 382 0155
support@endace.com
www.endace.com

Europe, Middle East & Africa
Endace Europe® Ltd
Sheraton House
Castle Park
Cambridge CB3 0AX
United Kingdom
Phone: ++44 1223 370 176
Fax: ++44 1223 370 040
support@endace.com
www.endace.com

Typographical Conventions Used in this Document

- Command-line examples suitable for entering at command prompts are displayed in `mono-space courier font`. The font is also used to describe config file data used as examples within a sentence. An example can be in more than one sentence.

  Results generated by example command-lines are also displayed in `mono-space courier font.`

- The software version references such as 2.3.x, 2.4.x, 2.5.x are specific to Endace Measurement Systems and relate to Company software products only.

Protection Against Harmful Interference

When present on product this manual pertains to and indicated by product labelling, the statement "This device complies with part 15 of the FCC rules" specifies the equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the Federal Communications Commission [FCC] Rules.

These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment.

This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications.

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

Extra Components and Materials

The product that this manual pertains to may include extra components and materials that are not essential to its basic operation, but are necessary to ensure compliance to the product standards required by the United States Federal Communications Commission, and the European EMC Directive. Modification or removal of these components and/or materials, is liable to cause non compliance to these standards, and in doing so invalidate the user’s right to operate this equipment in a Class A industrial environment.

Table of Contents

1.0 PREFACE..1
1.1 User Manual Purpose 1
1.2 DAG 3.6E Card Product Description...2
1.3 DAG 3.6E Card Series Architecture 2
1.3.1 DAG 3.6EP Card Architecture 3
1.3.2. DAG 3.6ET Card Architecture 4
2.0 INSTALLING DAG 3.6E CARD 5
2.1 Insert DAG 3.6E Card into PC..5
2.2 Connecting DAG Card Ports...6
2.2.1 Connect DAG 3.6EP Card Ports...6
2.2.2 Connect DAG 3.6ET Card Ports...7
2.2.3 Timing Synchronization...8
3.0 CONFIDENCE TESTING DAG 3.6E CARD...9
3.1 DAG Cards Sensitivity..9
3.1.1 DAG 3.6 EP Card Sensitivity 9
3.1.2 DAG 3.6ET Card Sensitivity 9
3.2 Interpreting DAG 3.6E Card LED Status 10
3.2.1 DAG 3.6EP Card LED Status...11
3.2.2 DAG 3.6ET Card LED Status...11
3.3 DAG Card Capture Session 12
3.4 Configuration in WYSYCC style...14
3.5 Inspect Interface Statistics...15
3.6 Reporting Problems...17
4.0 RUNNING DAG CARD DATA CAPTURE SOFTWARE 19
4.1 Starting DAG Card Capture Session..19
4.2 High Load Performance 21
5.0 SYNCHRONIZING CLOCK TIME...23
5.1 Configuration Tool Usage..24
5.2 Time Synchronization Configurations 25
5.2.1 Single Card no Reference Time Synchronization......................................25
5.2.2 Two Cards no Reference Time Synchronization.......................................26
5.2.3 Card with Reference Time Synchronization..27
5.3 Synchronization Connector Pin-outs..29
6.0 DATA FORMATS OVERVIEW..31
6.1 Data Formats 31
6.2 Timestamps 33

1.0 PREFACE

Introduction

The installation of the Endace DAG 3.6E card on a PC begins with installing the operating system and the Endace software. This is followed by fitting the card and connecting the ports.

The characteristics include card architecture, extended functions, and system requirements.

Viewing this document

This document, DAG 3.6E Card User Manual, is available on the installation CD.

In this chapter

This chapter covers the following sections of information.

- User Manual Purpose
- DAG 3.6E Card Product Description
- DAG 3.6E Card Series Architecture
- System Requirements

1.1 User Manual Purpose

Description

The purpose of this DAG 3.6E Card User Manual is to describe:

- Installing DAG 3.6E Card
- Confidence Testing DAG 3.6E Card
- Running DAG Card Data Capture Software
- Synchronizing Clock Time
- Data Formats Overview

Pre-requisite

This document presumes the DAG card is being installed in a PC already configured with an operating system.

A copy of the Debian Linux 3.1 (Sarge) is available as a bootable ISO image on one of the CD's shipped with the DAG card.

To install on the Linux/FreeBSD operating system, follow the instructions in the document EDM04.05-01r1 Linux FreeBSD Installation Manual, packaged in the CD shipped with the DAG card.

To install on a Windows operating system, follow the instructions in the document EDM04.05-02r1 Windows Installation Manual, packaged in the CD shipped with the DAG card

1.2 DAG 3.6E Card Product Description

The DAG Ethernet port will operate in half duplex or full duplex modes. The DAG 3.6E card by default finds the fastest link configuration possible with the peer device using Ethernet Autonegotiation.

Figure

Figure 1-1 shows a DAG 3.6E series card.

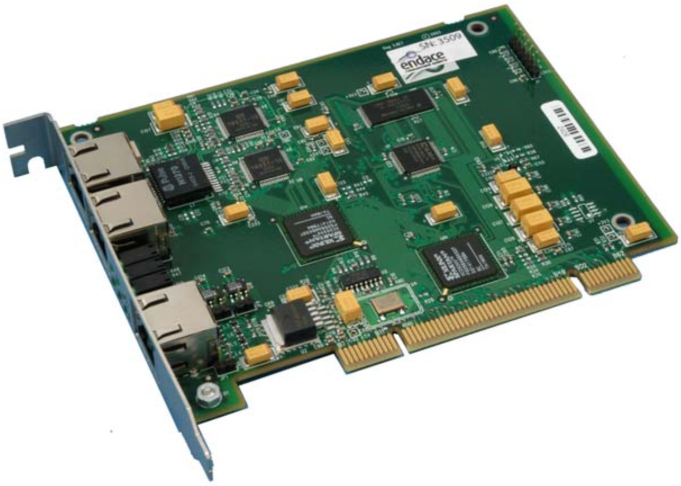

Figure 1-1. A DAG 3.6E series card.

1.3 DAG 3.6E Card Series Architecture

Description

The DAG 3.6E series of PCI-bus card consist of:

- DAG 3.6EP Dual Port 10Base-T/100Base-TX Ethernet interface card
- DAG3.6 ET full duplex 10Base-T/100Base-TX Ethernet Tap interface card

Continued on next page

1.3 DAG 3.6E Card Series Architecture, continued

Figure

Figure 1-2 shows the DAG 3.6E card series major components and process flow.

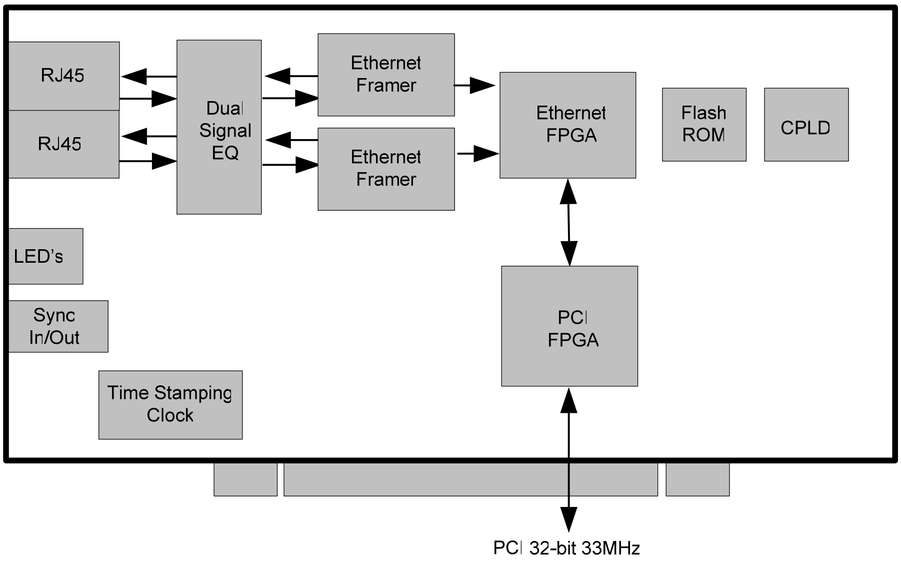

Figure 1-2. DAG 3.6E Card Series Major Components and Process Flow.

In this section

This section covers the following topics of information.

- DAG 3.6EP Card Architecture
- DAG 3.6ET Card Architecture

1.3.1 DAG 3.6EP Card Architecture

Description

The DAG 3.6EP has two independent 10/100 Ethernet interfaces.

Each can be connected to a separate switch or hub, and autonegotiate with the connected equipment.

Ethernet framers

Each single half or full-duplex 10 or 100Mbps Ethernet connection passes into a port of the DAG card. Two Ethernet framers look at data from each port independently. The DAG card captures from half duplex or full duplex links.

Packet time stamping

Serial Ethernet data is received by the interface, and fed through a framer into the upper of two Xilinx FPGA's. This FPGA contains an Ethernet processor and the DUCK timestamp engine.

Because of component close association, packets or cells are time-stamped accurately. Time stamped packet records are then stored in the lower FIFO.

Continued on next page

1.3.1 DAG 3.6EP Card Architecture, continued

Records transfer

Records transfer from FIFO into lower FPGA, which has interfaces to the PCI bus and then all records are written to host memory.

1.3.2. DAG 3.6ET Card Architecture

Description

The DAG 3.6ET is designed for tapping both directions of a single copper 10 or 100Mbps Ethernet link in a completely passive and fail-safe manner.

The DAG card will capture from half duplex or full duplex links. The DAG card does not interfere in the link negotiation process in any way, so full duplex links will remain full duplex after being connected through the DAG card.

Ethernet framers

A single half or full-duplex 10 or 100Mbps Ethernet connection passes into one port of the DAG card, and out the other un-interrupted.

Two Ethernet framers look at data travelling in each direction on the connection without interfering with the link. The DAG card cannot transmit onto this link at all, and even if the DAG 3.6ET is turned off the network link integrity is assured.

Packet time stamping

Serial Ethernet data is received by the interface, and fed through a framer into the upper of two Xilinx FPGA's. This FPGA contains an Ethernet processor and the DUCK timestamp engine.

Because of component close association, packets are time-stamped accurately. Time stamped packet records are then stored in a FIFO.

Records transfer

Records transfer from FIFO into lower FPGA, which has interfaces to the PCI bus and then all records are written to host memory.

Description

The DAG 3.6E and associated data capture system minimum operating requirements are:

- PC, at least Pentium II 400 MHz, Intel 440BX, GX or newer chip set
- 256 MB RAM
- At least one free PCI free slot with 3.3V and 5V power
- Software distribution requires free space of 30MB

Operating system

For convenience, a Debian 3.1 [Sarge] Linux system is included on the Endace Software Install CD. Endace currently supports Windows XP, Windows Server 2000, Windows Server 2003, FreeBSD, RHEL 3.0, and Debian Linux operating systems.

Different system

For advice on using a system substantially different from that specified above, contact Endace support at support@endace.com

2.0 INSTALLING DAG 3.6E CARD

Introduction

A DAG 3.6E card can be installed in any free Bus Mastering PCI slot.

Although the driver supports up to four DAG cards by default in one system, due to band width limitations there should not be more than two cards on a single PCI-bus.

In this chapter

This chapter covers the following sections of information.

- Insert DAG 3.6E Card into PC
- Connecting DAG Card Ports

2.1 Insert DAG 3.6E Card into PC

Description

Inserting the DAG 3.6E card into a PC involves accessing the bus slot, fitting the card, and replacing bus slot cover.

Procedure

Follow these steps to insert the DAG 3.6E card.

Step 1. **Access bus Slot**

Power computer down.

Remove PCI-bus slot cover.

Step 2. **Fit Card**

Insert into PCI-bus slot.

Step 3. **Replace bus Slot Screw**

Secure card with screw.

Step 4. **Power up Computer**

2.2 Connecting DAG Card Ports

Description The DAG 3.6EP and DAG ET card ports are for ethernet connection, and one port for time synchronisation.

In this section This section covers the following topics of information.

- Connect DAG 3.6EP Card Ports
- Connect DAG 3.6ET Card Ports
- Timing Synchronization

2.2.1 Connect DAG 3.6EP Card Ports

Description With the DAG 3.6EP there are two RJ45 ethernet ports that can be connected independently to any two Ethernet devices.

Typically the DAG card is connected to two hub, switch, or router ports.

Figure Figure 2-1 shows the typical DAG card connection to two hub, switch, or router ports.

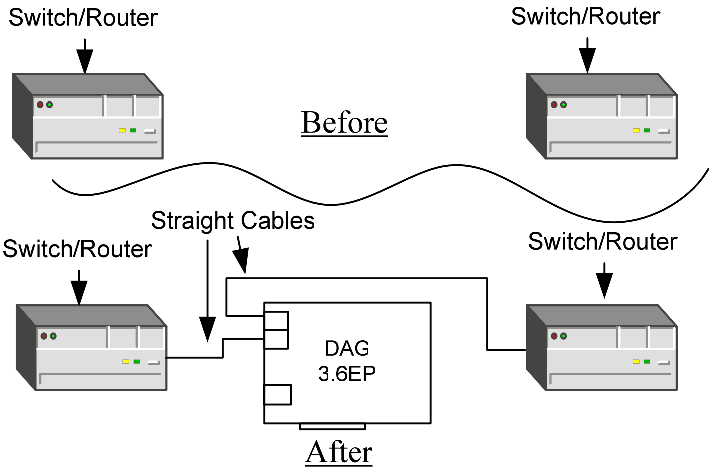

Figure 2-1. Typical DAG card connection to Hub, Switch, or Router Ports.

Pin jumpers There are a number of pin jumpers on the DAG 3.6EP card. These are set when a card is manufactured, and must never be altered.

Voiding warranty Changing jumper settings can cause the card to permanently malfunction and may void the product warranty

2.2.2 Connect DAG 3.6ET Card Ports

Description

The two DAG 3.6ET RJ45 ports should be connected in series with an existing ethernet link. Typically, the DAG card is connected between a NIC and a Switch,

Figure

Figure 2-2 shows the DAG card connection between a NIC and switch.

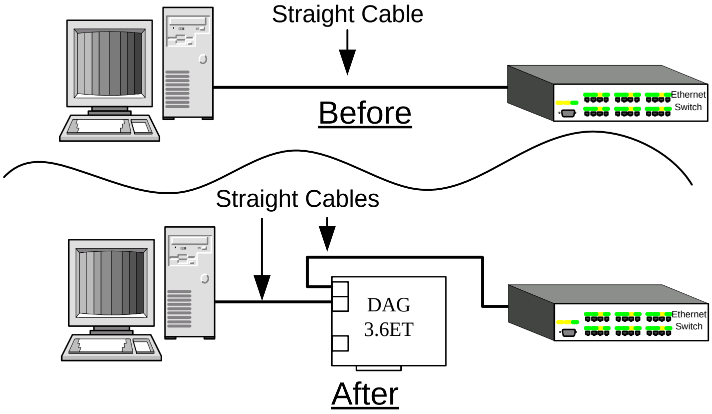

Figure 2-2. Typical DAG Card Connection Between a NIC and Switch.

Insertion between switches and routers

Sometimes the DAG card may be inserted between two switches or routers. In such cases, a cross-over Ethernet cable is required. The Router directly connected can be referred to as the 'Switch' end of the connection, and the Router connected via the cross-over cable can be referred to as the 'NIC' end of the connection.

Figure

Figure 2-3 shows a typical DAG card insertion between two switches or routers.

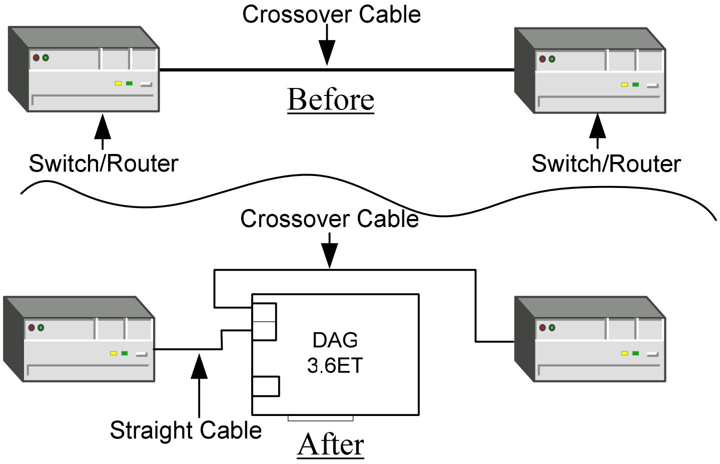

Figure 2-2. Typical DAG Card Insertion Between Two Switches or Routers.

Continued on next page

2.2.2 Connect DAG 3.6ET Card Ports, continued

Cross-over cables description

If the DAG card is inserted between two NIC cards, then one cross-over cable is needed, as shown in Figure 2-2.

In such case the :

- Directly connected NIC is referred to as the ‘NIC’
- NIC connected by cross-over cable is referred to as the ‘Switch’

Port A and Port B on the DAG 3.6ET are interchangeable; it does not matter which of the ports each cable is connected to.

Pin jumpers

There are a number of pin jumpers on the DAG 3.6EP card. These are set when a card is manufactured, and must never be altered.

Voiding warranty

Changing jumper settings can cause the card to permanently malfunction and may void the product warranty

2.2.3 Timing Synchronization

Description

The DAG 3.6E has a third RJ45 socket near the PCI connector for time synchronization input. This port should never be connected to an Ethernet network or telephone line.

3.0 CONFIDENCE TESTING DAG 3.6E CARD

Introduction

The confidence testing is a process to determine the DAG 3.6E card series is functioning relative to the LED functions status. The process also involves a card capture session, and configuration in the style of what you can see you can change.

The interface statistics are inspected. This section also has information regarding reporting of problems.

In this chapter

This chapter covers the following sections of information.

- DAG Cards Sensitivity
- Interpreting DAG 3.6E Card LED Status
- DAG Card Capture Session
- Configuration in WYSYCC style
- Inspect Interface Statistics
- Reporting Problems

3.1 DAG Cards Sensitivity

In this section

This section covers the following topics of information.

- DAG 3.6 EP Card Sensitivity
- DAG 3.6ET Card Sensitivity

3.1.1 DAG 3.6 EP Card Sensitivity

Description

The DAG 3.6EP card is compliant to IEEE 802.3 standards in respect to maximum cable lengths, and operates as a normal Ethernet device. Activity lights indicate network traffic in each direction on the link.

3.1.2 DAG 3.6ET Card Sensitivity

Description

The insertion loss associated with inserting a DAG 3.6ET card into an existing link is <1dB. With this low insertion loss, the DAG 3.6ET can be inserted into long cable runs.

By default the DAG 3.6ET will attempt to automatically detect the line speed of either 10 or 100Mbps, and light link status lights to indicate that it has detected the network correctly.

Activity lights indicate network traffic in each direction on the link.

The DAG 3.6ET card cannot introduce traffic onto the network link, and the link will remain up even if the DAG 3.6ET card is misconfigured or powered off.

3.2 Interpreting DAG 3.6E Card LED Status

Description

The dagthree utility supports configuration status and physical layer interface statistics for the DAG 3.x series of cards.

In a troubleshooting configuration options –si should be passed to the tool to watch physical and framing layers operational status.

More details about the meaning of the various bits are supplied through the help page (`dagthree -h`) as well as via the manual page.

Status LED's

The DAG 3.6E has 12 status LEDs, three are coloured orange, and nine green.

Figure

Figure 3-1 shows the DAG 3.6E status LED's.

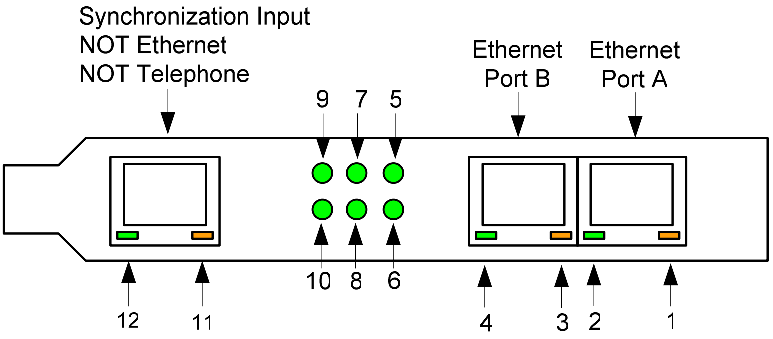

Figure 3-1. DAG 3.6E Status Lights.

In this section

This section covers the following topics of information.

- DAG 3.6 EP Card Sensitivity
- DAG 3.6ET Card Sensitivity

3.2.1 DAG 3.6EP Card LED Status

LED definitions The DAG 3.6EP card LED function definitions are:

LED	Description
LED 1	Link Activity Port A
LED 2	Link Up Port A
LED 3	Link Activity Port B
LED 4	Link Up Port B
LED 5	PCI (Lower) FPGA successfully programmed
LED 6	PP (Upper) FPGA successfully programmed
LED 7	Burst Manager Run; Indicates the card is capturing packets and transferring them to the host
LED 8	Port mode. 3.6 EP Always On, 3.6 ET Always Off
LED 9	Port A 100Base-TX, off for 10Base-T
LED 10	Port B 100Base-TX, off for 10Base-T
LED 11	PPS Out: Pulse Per Second Out – indicates the card is sending a clock synchronisation signal
LED 12	PPS In: Pulse Per Second In – indicates the card is receiving an external clock synchronisation signal

3.2.2 DAG 3.6ET Card LED Status

Description The DAG 3.6EP card LED function definitions are:

LED 1	The ‘Switch’ device is transmitting
LED 2	Link detected from ‘Switch’
LED 3	The ‘NIC’ device is transmitting
LED 4	Link detected from ‘NIC’
LED 5	PCI (Lower) FPGA successfully programmed
LED 6	PP (Upper) FPGA successfully programmed
LED 7	Burst Manager Run – Indicates the card is capturing packets and transferring them to the host
LED 8	Reserved
LED 9	Switch’ 100Base-TX, off for 10Base-T
LED 10	‘NIC’ 100Base-TX, off for 10Base-T
LED 12	PPS Out: Pulse Per Second Out – indicates the card is sending a clock synchronisation signal
LED 13	PPS In: Pulse Per Second In – indicates the card is receiving an external clock synchronisation signal

3.3 DAG Card Capture Session

Description

The DAG 3.6E uses two ASIC framer devices to support capturing of Ethernet data frames. The card supports both 10Base-T and 100Base-TX standards.

The default behaviour is to autodetect the correct link speed, however the configuration can be forced if necessary.

A successful DAG 3.6E card capture session is accomplished by receiver ports optical signal levels and checking the card is locked to stream data. This is followed by configuring DAG for normal use.

Procedure

Follow these steps for a successful DAG 3.6E card capture session.

Step 1. Check Cabling

Ensure cabling is correctly connected and that RJ45 connectors are clipped into the sockets.

Step 2. Check link layer configuration

Learn about the link layer configuration in use at the network link being monitored. Important parameters include:

- 10Base-T vs. 100Base-TX configuration
- Full vs. Half Duplex.

If the information cannot be obtained reliably, the card can be made to work by varying parameters until data is arriving at the host system.

Step 3. List Current Settings

For DAG 3.6E framer configuration and statistics the `dagthree` tool is supplied.

Calling `dagthree` without arguments lists current settings.

The `dagthree -h` prints a help listing on tool usage.

```
dag@endace:~$ dagthree -d dag0
linkA   noreset tap auto enableA
linkB   noreset tap auto enableB
packet  novarlen slen=48 noalign64
packetA drop=0
packetB drop=0
pci     33MHz 32-bit buf=32MB rxstreams=1 txstreams=0 mem=0:0
```

Continued on next page

3.3 DAG Card Capture Session, continued

Procedure (continued)

Step 4. **Check FPGA Images Loaded**

Before starting to configure the card, ensure the most recent pair of FPGA images have been loaded onto the card.

The link status and activity LEDs will not activate until the upper FPGA firmware is downloaded.

```
dag@endace:~$ dagrom -rvp –d dag0 -f xilinx/dag36epci-erf.bit

dag@endace:~$ dagld -x –d dag0 –f xilinx/dag36epp-erf.bit
```

Step 5. **Configure Card for Normal Use**

Configure according to local settings using `dagthree default` command.

NOTE: If `dagthree default` command is not used the card will operates correctly but mem display will show 0 memory allocated to the receive stream buffer until after the first capture.

```
dag@endace:~$ dagthree default
linkA   noreset tap auto enableA
linkB   noreset tap auto enableB
packet  novarlen slen=48 noalign64
packetA drop=0
packetB drop=0
pci     33MHz 32-bit buf=32MB rxstreams=1 txstreams=0 mem=32:0
```

Step 6. **Check Card Operation**

Check through the physical layer statistics that the card is locked to the data stream.

3.4 Configuration in WYSYCC style

Description

Configuration in WYSYCC is the 'What You See You Can Change' style.

Running the command `dagthree` alone shows the current configuration. Each of the items displayed can be changed as follows:

Procedure

Follow this step to set variable length capture configuration.

Step 1. Configure Variable Length Capture.

Type:

```
dag@endace:~$ dagthree -d dag0 varlen
linkA   noreset tap auto enableA
linkB   noreset tap auto enableB
packet  varlen slen=48 noalign64
packetA drop=0
packetB drop=0
pci     33MHz 32-bit buf=32MB rxstreams=1 txstreams=0 mem=32:0
```

NOTE: The card status will include `tap` for each link line for the DAG 3.6ET, and will contain `port` for each link line for the DAG 3.6EP. These values are not software adjustable.

Step 2. Select Configuration Option

Choose from complete list of configuration options supported:

`reset`	Reset Ethernet framers, set auto mode.
`default`	Reset Ethernet framers, set auto mode.
`auto`	Set autonegotiate mode, card detects rate.
`10`	Force 10BaseT mode,10Mbps.
`100`	Force 10BaseT mode,100Mbps.
`[en|dis]ableA`	Enable or disable port A direction for capture.
`[en|dis]ableB`	Enable or disable port B direction for capture.
`[no]varlen`	Dis/enable variable length capture. Otherwise record length padded to slen.
`slen=`	Capture X bytes of packet content.
`[no]align64`	Generate records with 64-bit alignment [default 32-bit]

Step 3. Force DAG 3.6E to 10Mbps Mode.

While monitoring a Mbps connection and autodetection fails to switch to 10Mbps, use

`dagthree 10` to force DAG 3.6E to 10Mbps mode.

3.5 Inspect Interface Statistics

Description

Once the card has been configured, the interface statistics are inspected to check the card is locked to the data stream.

```
dag@endace:~$ dagthree -d dag0 -si
```

The tool displays a number of status bits with their current values. In the following example, the interval is set to one second via the `-i` option. The –a and –b options cause only the status for a single port to be displayed, otherwise both are shown.

`Spd`	Link Speed, 10 or 100 Mbps
`SD`	Signal Detect [100Mbps only]
`Lck`	PLL Lock [100Mbps only]
`Neg`	Auto-negotiation completed [Auto mode only]
`FC`	False Carrier Detected Error
`IS`	Invalid Symbol Detected Error
`HS`	Halt Symbol Detected Error
`PE`	Premature End Detected Error
`RF`	Remote Fault Detected Error
`JB`	Jabber Detected Error

Port A card with no valid input.

The following is an example for the Port A card with no valid input.

```
dag@endace:~$ dagthree -d dag0 –asi
```

```
A:   Spd    SD   Lnk    Lck    Neg    FC    IS    HS    PE    RF    JB
      10     0     0      0      0     0     0     0     0     0     0
      10     0     0      0      0     0     0     0     0     0     0
      10     0     0      0      0     0     0     0     0     0     0
      10     0     0      0      0     0     0     0     0     0     0
```

Port A card locked to 10Base-T stream

The following is an example for Port A of a card locked to a 10Base-T stream.

```
dag@endace:~$ dagthree -d dag0 -asi
```

```
A:   Spd    SD   Lnk    Lck    Neg    FC    IS    HS    PE    RF    JB
      10     0     1      0      1     0     0     0     0     0     0
      10     0     1      0      1     0     0     0     0     0     0
      10     0     1      0      1     0     0     0     0     0     0
      10     0     1      0      1     0     0     0     0     0     0
```

Continued on next page

3.5 Inspect Interface Statistics, continued

Port A card locked to 100Base-T stream

The following is an example for Port A of a card locked to a 100Base-T stream.

```
dag@endace:~$ dagthree -d dag0 -asi

A:   Spd    SD   Lnk   Lck   Neg   FC   IS   HS   PE   RF   JB
     100     1     1     1     1    0    0    0    0    0    0
     100     1     1     1     1    0    0    0    0    0    0
     100     1     1     1     1    0    0    0    0    0    0
     100     1     1     1     1    0    0    0    0    0    0
```

Network link fault corrective action

A network link fault problem is indicated if any other bits are numeral 1. This may or may not be related to addition of the DAG 3.6E card.

All cabling is checked, ensuring that runs are not too long and plugs firmly clipped into connectors.

Error condition detectors or counters on the Ethernet equipment are checked.

Corrective action can also be performed by removing the DAG card from the link.

3.6 Reporting Problems

Description

Support is provided with a service contract. If problems with a DAG card or supplied software, contact Endace Technical Support via the email address support@endace.com. Supplying sufficient information in an email enables effective response.

Problem checklist

The exact information available to users for trouble, cause and correction analysis may be limited by nature of the problem. The following items assist a quick problem resolution:

Ref **Item**

1. DAG card[s] model and serial number.
2. Host PC type and configuration.
3. Host PC operating system version.
4. DAG software version package in use.
5. Any compiler errors or warnings when building DAG driver or tools.
6. For Linux and FreeBSD, messages generated when DAG device driver is loaded. These can be collected from command `dmesg`, or from log file `/var/log/syslog`.
7. Output of `daginf`.
8. Firmware version `dagrom –x`.
9. Physical layer status reported by:

   ```
   dagthree
   ```

10. Network link statistics reported by:

    ```
    dagthree –si
    ```

11. Network link configuration from the router where available.
12. Contents of any scripts in use.
13. Complete output of session where error occurred including any error messages from DAG tools. The typescript Unix utility may be useful for recording this information.
14. A small section of a captured packet trace illustrating the problem.

USE THIS SPACE FOR NOTES

4.0 RUNNING DAG CARD DATA CAPTURE SOFTWARE

Introduction

For a typical measurement session, the `scripts/dag36estart` script is edited and used to operate the cards directly.

In this chapter

This chapter covers the following sections of information.

- Starting DAG Card Capture Session
- High Load Performance

4.1 Starting DAG Card Capture Session

Description

Capture settings are set for both cards in use. The various tools used for data capture are in the `tools` sub-directory.

For a typical measurement session, first move to the `dag` directory, load the driver, then load the Xilinx receive image to each DAG. For example, with one DAG 3.6E card installed:

```
tools/dagrom -rvp –d dag0 -f xilinx/dag36epci-erf.bit
tools/dagld -x –d dag0 –f xilinx/dag36epp-erf.bit
```

The integrity of both DAG cards physical layer is then set and checked.

Process

Starting a data capture session is described in the following process.

Process	Description

Load directory.	Move to `dag` directory and load driver, and load Xilinx receive image to each DAG.
	For example, with one DAG 3.6E card installed:
	`tools/dagrom -rvp –d dag0 –f xilinx/dag36epci-erf.bit`
	`tools/dagld -x –d dag0 –f xilinx/dag36epp-erf.bit`

Set physical layer integrity.	Set and check the integrity of the physical layer of the DAG LED status.

Continued on next page

4.1 Starting DAG Card Capture Session, continued

Process (continued)

Process	Description

Set session parameters.	The card can operate in two modes:
	Variable length capture (`varlen`)
	Fixed length capture (`novarlen`).
	NOTE: In variable length capture mode, a maximum capture size is set with `slen=N` bytes. This figure should be in the range 32 to 2048 and is rounded down to nearest multiple of 4. Packets longer then slen will be truncated. Packets shorter than slen produce shorter records, saving bandwidth and storage space. An example for full packet capture is:
	`dagthree –d dag0 varlen slen=1536`
	NOTE: In fixed length mode, packets longer than selected slen truncate to slen, but packets shorter than slen produce records padded out to value of slen. For this reason large values of slen are not used in fixed length mode, as short packets arriving will produce large padded records, wasting bandwidth and storage space.
	For example, for fixed length 64-byte records, choose slen=44 (64 – ERF header size of 16 – alignment padding 4):
	`dagthree –d dag0 novarlen slen=44`

Setting port signal values.	Each direction [A and B] can be individually enabled and disabled for capture using `dagthree`. `dagthree –d dag0 disableb` Packets that arrive on a disabled direction are not counted as lost.

Continued on next page

4.1 Starting DAG Card Capture Session, continued

Process (continued)

Process	Description
Starting capture session.	Started on a card using: `dagsnap –v –o tracefile` NOTE: Option `-v` provides user information during capture; it can be omitted for automated trace runs. If the `tracefile` parameter is not specified the tool writes to stdout, which can be used to pipeline `dagsnap` with other tools from `dagtools` package.
Stopping `dagsnap`	By default `dagsnap` will run forever. `dagsnap` can be stopped with a signal: `killall dagsnap` `dagsnap` can also be configured to run for a fixed number of seconds and then exit with the `–s` option

4.2 High Load Performance

Description

As the DAG card captures packets from the network link, it writes a record for each packet into a large buffer in the host PC’s main memory.

Avoiding packet loss

In order to avoid packet loss, the user application reading the record, such as `dagsnap`, must be able to read records out of the buffer faster than they arrive, otherwise the buffer eventually fills, and packet records are lost.

The “Data capture” LED also goes out. This may be visibly indicated as flashing or flickering.

Detecting packet losses

Until some data is read out of the buffer to free some space, any arriving packets subsequently are discarded by the DAG card.

Any loss can be detected in-band by observing the Loss Counter `lctr` field of the Extensible Record Format. The Endace ERF is detailed in Chapter 7 of this document.

Continued on next page

4.2 High Load Performance, continued

Avoiding packet loss

In order to avoid any potential packet loss, the user process must read records faster than they arrive from the network.

For Linux and FreeBSD, when the PC buffer becomes full, the message:

```
kernel: dagN: pbm safety net reached
```

is displayed on the PC screen, and printed to `log /var/log/messages`.

If the user process is writing records to hard disk, it may be necessary to use a faster disk or disk array. If records are being processed in real-time, a faster host CPU may be required.

Increasing buffer size

The host PC buffer can be increased to deal with bursts of high traffic load on the network link.

By default the dagmem driver reserves 32MB of memory per DAG card in the system. For OC-12/STM-4 962Mbps) rates and above. This may require 128MB or more for Linux/FreeBSD and for the Windows operating system the requirement is 64MB or more.

128MB or more is suggested for Linux/FreeBSD.

For the DAG 3.6E card Windows operating system the upper limit is 32MB.

In Debian Linux the amount of memory reserved is changed by editing the file `/etc/modules`.

```
# For DAG 3.x, default 32MB/card
dagmem
#
# For DAG 4.x or 6.x, use more memory per card, E.G.
# dagmem dsize=128m
```

The option `dsize` sets the amount of memory used per DAG card in the system.

The value of `dsize` multiplied by the number of DAG cards must be less than the amount of physical memory installed, and less than 890MB.

5.0 SYNCHRONIZING CLOCK TIME

Description

The Endace DAG range of products come with sophisticated time synchronisation capabilities, in order to provide high quality timestamps, optionally synchronized to an external time standard.

The system that provides the DAG synchronisation capability is known as the DAG Universal Clock Kit (DUCK).

An independent clock in each DAG card runs from the PC clock. A card’s clock is initialised using the PC clock, and then free-runs using a crystal oscillator.

Each card's clock can vary relative to a PC clock, or other DAG cards.

DUCK configuration

The DUCK is configured to avoid time variance between sets of DAG cards or between DAG cards and coordinated universal time [UTC].

Accurate time reference can be obtained from an external clock by connecting to the DAG card using the synchronisation connector, or the host PCs clock can be used in software as a reference source without additional hardware.

Each DAG card can also output a clock signal for use by other cards.

Common synchronization

The DAG card synchronisation connector supports a Pulse-Per-Second (PPS) input signal, using RS-422 signalling levels.

Common synchronisation sources include GPS or CDMA (Cellular telephone) time receivers.

Endace produces the TDS 2 Time Distribution Server modules and the TDS 6 units that enable multiple DAG cards to be connected to a single GPS or CDMA unit.

More information is on the Endace website, http://www.endace.com/accessories.htm, or the TDS 2/TDS 6 Units Installation Manual.

In this chapter

This chapter covers the following sections of information.

- Configuration Tool Usage
- Time Synchronization Configurations
- Synchronization Connector Pin-outs

5.1 Configuration Tool Usage

Description

The DUCK is very flexible, and can be used in several ways, with or without an external time reference source. It can accept synchronisation from several input sources, and can also be made to drive its synchronisation output from one of several sources.

Synchronisation settings are controlled by the `dagclock` utility.

Example

```
dag@endace:~$ dagclock -h
Usage: dagclock [-hvVxk] [-d dag] [-K <timeout>] [-l
<threshold>] [option]

     -h  --help,--usage   this page
     -v  --verbose        increase verbosity
     -V  --version        display version information
     -x  --clearstats     clear clock statistics
     -k  --sync           wait for duck to sync before
                          exiting
     -d  dag              DAG device to use
     -K  timeout          sync timeout in seconds, default
                          60
     -l  threshold        health threshold in ns, default
                          596
  Option:
         default           RS422 in, none out
         none              None in, none out
         rs422in           RS422 input
         hostin            Host input (unused)
         overin            Internal input (synchronize to
                           host clock)
         auxin             Aux input (unused)
         rs422out          Output the rs422 input signal
         loop              Output the selected input
         hostout           Output from host (unused)
         overout           Internal output (master card)
         set               Set DAG clock to PC clock
         reset             Full clock reset.  Load time
                           from PC, set rs422in, none out
```

By default, all DAG cards listen for synchronisation signals on their RS-422 port, and do not output any signal to their RS-422 port.

```
dag@endace:~$ dagclock –d dag0
muxin   rs422
muxout  none
status  Synchronized Threshold 596ns Failures 0 Resyncs 0
error   Freq -30ppb Phase -60ns Worst Freq 75ppb Worst
Phase 104ns
crystal Actual 100000028Hz Synthesized 67108864Hz
input   Total 3765 Bad 0 Singles Missed 5 Longest
Sequence Missed 1
start   Thu Apr 28 13:32:45 2005
host    Thu Apr 28 14:35:35 2005
dag     Thu Apr 28 14:35:35 2005
```

5.2 Time Synchronization Configurations

Description

The DUCK is very flexible, and can be used in several ways, with or without an external time reference source.

The use includes a single card with no reference, two cards with no reference, and a card with reference.

In this section

This section covers the following topics of information.

- Single Card no Reference Time Synchronization
- Two Cards no Reference Time Synchronization
- Card with Reference Time Synchronization

5.2.1 Single Card no Reference Time Synchronization

Description

When a single card is used with no external reference, the card can be Synchronized to the host PC's clock.

The clock in most PC's is not very accurate by itself, but the DUCK drifts smoothly at the same rate as the PC clock.

If a PC is running NTP to synchronize its own clock, then the DUCK clock is less smooth because the PC clock is adjusted in small jumps. However, overall the DUCK clock does not drift away from UTC.

The synchronisation achieved in this case is not as accurate as when using an external reference source such as GPS.

The DUCK clock is synchronized to a PC clock by setting input synchronization selector to overflow:

```
dag@endace:~$ dagclock –d dag0 none overin
muxin   overin
muxout  none
status  Synchronized Threshold 11921ns Failures 0 Resyncs
0
error   Freq 1836ppb Phase 605ns Worst Freq 143377ppb
Worst Phase 88424ns
crystal Actual 49999347Hz Synthesized 16777216Hz
input   Total 87039 Bad 0 Singles Missed 0 Longest
Sequence Missed 0
start   Wed Apr 27 14:27:41 2005
host    Thu Apr 28 14:38:20 2005
dag     Thu Apr 28 14:38:20 2005
```

NOTE: `dagclock` should be run only after appropriate Xilinx images have been loaded. If the Xilinx images must be reloaded, the `dagclock` command must be rerun afterwards to restore the configuration.

5.2.2 Two Cards no Reference Time Synchronization

Description

When two DAG cards are used in a single host PC with no reference clock, the cards are to be synchronized in some way if timestamps between the two cards are to be compared. For example, if two cards monitor different directions of a single full-duplex link.

Synchronization between two DAG cards is achieved in two ways. One card can be a clock master for the second, or one can synchronize to the host and also act as a master for the second.

Synchronizing cards

If both cards are to be accurately Synchronized, then one card is configured as the clock master for the other.

Locking cards together

Although the master card’s clock will drift against UTC, the cards are locked together.

The cards are locked together by connecting the synchronisation connector ports of both cards with a standard RJ-45 Ethernet cross-over cable.

Configure one of the cards as the master, the other defaults to being a slave.

```
dag@endace:~$ dagclock –d dag0 none overout
muxin   none
muxout  over
status  Not Synchronized Threshold 596ns Failures 0
Resyncs 0
error   Freq 0ppb Phase 0ns Worst Freq 0ppb Worst Phase
0ns
crystal Actual 100000000Hz Synthesized 67108864Hz
input   Total 0 Bad 0 Singles Missed 0 Longest Sequence
Missed 0
start   Thu Apr 28 14:48:34 2005
host    Thu Apr 28 14:48:34 2005
dag     No active input - Free running
```

The slave card configuration is not shown, the default configuration is sufficient.

Continued on next page

5.2.2 Two Cards no Reference Time Synchronization, continued

Preventing time-stamps drift

To prevent the DAG card clocks time-stamps drifting against UTC, the master can be Synchronized to the host PC's clock which in turn utilises NTP. This then provides a master signal to the slave card.

The cards are locked together by connecting the synchronisation connector ports of both cards with a standard RJ-45 Ethernet cross-over cable.

Configure one card to synchronize to the PC clock and output a RS-422 synchronization signal to the second card.

```
dag@endace:~$ dagclock –d dag0 none overin overout
muxin   over
muxout  over
status  Synchronized Threshold 11921ns Failures 0 Resyncs
0
error   Freq -691ppb Phase -394ns Worst Freq 143377ppb
Worst Phase 88424ns
crystal Actual 49999354Hz Synthesized 16777216Hz
input   Total 87464 Bad 0 Singles Missed 0 Longest
Sequence Missed 0
start   Wed Apr 27 14:27:41 2005
host    Thu Apr 28 14:59:14 2005
dag     Thu Apr 28 14:59:14 2005
```

The slave card configuration is not shown, the default configuration is sufficient.

5.2.3 Card with Reference Time Synchronization

Description

The best timestamp accuracy occurs when DAG card is connected to an external clock reference, such as a GPS or CDMA time receiver.

Pulse signal from external sources

The DAG synchronisation connector accepts a RS-422 Pulse Per Second [PPS] signal from external sources.

This is derived directly from a reference source, or distributed through the Endace TDS 2 [Time Distribution Server] module which allows two DAG cards to use a single receiver.

More cards can be accommodated by daisy-chaining TDS-6 expansion units to the TDS-2 unit, each providing outputs for an additional 6 DAG cards.

Continued on next page

5.2.3 Card with Reference Time Synchronization, continued

Using external reference source

To use an external clock reference source, the host PC's clock must be accurate to UTC to within one second. This is used to initialise the DUCK.

The external time reference allows high accuracy time synchronisation.

When the time reference source is connected to the DAG synchronisation connector, the card automatically synchronizes to a valid signal.

```
dag@endace:~$ dagclock –d dag0
muxin rs422
muxout none
status Synchronized Threshold 596ns Failures 0 Resyncs 0
error Freq 30ppb Phase -15ns Worst Freq 2092838ppb Worst
Phase 33473626ns
crystal Actual 100000023Hz Synthesized 67108864Hz
input Total 225 Bad 0 Singles Missed 1 Longest Sequence
Missed 1
start Thu Apr 28 14:55:20 2005
host Thu Apr 28 14:59:06 2005
dag Thu Apr 28 14:59:06 2005
```

Connecting time distribution server

The TDS 2 module connects to any DAG card with a standard RJ-45 Ethernet cable and can be placed some distance from a DAG card.

Existing RJ-45 building cabling infrastructure can be used to cable synchronisation ports.

CAUTION: Never connect DAG and/or the TDS 2 module to active Ethernet or telephone equipment.

Testing signal

For Linux and FreeBSD, when a synchronisation source is connected the driver outputs some messages to the console log file `/var/log/messages`.

The `dagpps` tool is used to test a signal is being received correctly and is of correct polarity. To perform the test, run:

```
dagpps –d dag0.
```

The tool measures input state many times over several seconds, displaying polarity and length of input pulse.

Some DAG cards have LED indicators for synchronisation (PPS) signals.

5.3 Synchronization Connector Pin-outs

Description

DAG cards have an 8-pin RJ45 connector with two bi-directional RS422 differential circuits, A and B. The PPS signal is carried on circuit A, and the serial packet is connected to the B circuit.

Pin assignments

The 8-pin RJ45 connector pin assignments are:

1.	Out A+
2.	Out A-
3.	In A+
4.	In B+
5.	In B-
6.	In A-
7.	Out B+
8.	Out B-

Figure

Figure 6-1 shows the RJ45 plug and socket connector pin-outs.

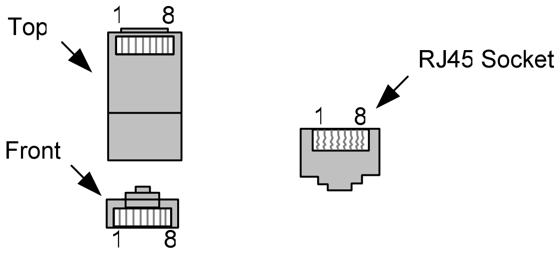

Figure 6-1. RJ45 Plug and Socket Connector Pin-outs.

Out-pin connections

Normally the GPS input should be connected to the A channel input, pins 3 and 6. The DAG can also output a synchronization pulse; used when synchronizing two DAG's without a GPS input. Synchronization output is generated on the Out A channel, pins 1 and 2.

Ethernet crossover cable

A standard Ethernet crossover cable can be used to connect the two cards.

TX_A+	1	3	RX_A+
TX_A-	2	6	RX-A-
RX_A+	3	1	TX_A+
RX_B+	4	7	TX_B+
RX_B-	5	8	TX_B-
RX_A-	6	2	TX_A-
TX_B+	7	4	RX_B+
TX_B-	8	5	RX_B-

Support

For cables and further advice on using GPS and CDMA time receivers email support@endace.com.

USE THIS SPACE FOR NOTES

6.0 DATA FORMATS OVERVIEW

In this chapter This chapter covers the following sections of information.

- Data Formats
- Timestamps

6.1 Data Formats

Description The DAG card uses the ERF Type 2 Ethernet Variable Length Record. Timestamps are in little-endian [Pentium native] byte order. All other fields are in big-endian [network] byte order. All payload data is captured as a byte stream, no byte re-ordering is applied.

Table Table 7-1 shows the generic variable length record. The diagram is not to scale.

<table>
<tr><td colspan="4">timestamp</td></tr>
<tr><td colspan="4">timestamp</td></tr>
<tr><td>type</td><td>flags</td><td colspan="2">rlen</td></tr>
<tr><td colspan="2">lctr</td><td colspan="2">wlen</td></tr>
<tr><td colspan="4">(rlen - 16) bytes of record</td></tr>
</table>

Table 7-1. Generic Variable Length Record.

Data format The following is an overview of the data format used.

Data Format	Description
type:	This field contains an enumeration of the frame subtype. If the type is zero, then this is a legacy format. 0: TYPE_LEGACY 1: TYPE_HDLC_POS: PoS w/HDLC framing 2: TYPE_ETH: Ethernet 3: TYPE_ATM: ATM Cell 4: TYPE_AAL5: reassembled AAL5 frame 5: TYPE_MC_HDLC: Multi-channel HDLC frame 6: TYPE_MC_RAW: Multi-channel Raw link data 7: TYPE_MC_ATM: Multi-channel ATM Cell

Continued on next page

6.1 Data Formats, continued

Data format, continued

Data Format	Description
flags:	This byte is divided into 2 parts, the interface identifier, and the capture offset. 1-0: capture interface 0-3 2: varying record lengths present 3: truncated record [insufficient buffer space] 4: rx error [link error] 5: 5: ds error [internal error] 7-6: reserved
Rlen: record length	Total length of the record transferred over PCI bus to storage.
Lctr: *loss counter*	A 16 bit counter, recording the number of packets lost since the previous record. Records can be lost between the DAG card and memory hole due to overloading on PCI bus. The counter starts at zero, and sticks at 0xffff.
Wlen: *wire length*	Packet length including some protocol overhead. The exact interpretation of this quantity depends on physical medium.
offset:	Number of bytes *not* captured from start of frame. Typically used to skip link layer headers when not required in order to save bandwidth and space. This field is currently not implemented, contents can be disregarded.

Continued on next page

6.1 Data Formats, continued

Table

Table 7-2 shows the Type 2 Ethernet variable length record. The diagram is not to scale.

<table>
<tr><td colspan="4">timestamp</td></tr>
<tr><td colspan="4">timestamp</td></tr>
<tr><td>type:2</td><td>flags</td><td colspan="2">rlen</td></tr>
<tr><td colspan="2">lctr</td><td colspan="2">wlen</td></tr>
<tr><td>offset</td><td>pad</td><td colspan="2">rlen-18</td></tr>
<tr><td colspan="4">bytes of frame</td></tr>
</table>

Table 7-2. Type 2 Ethernet Variable Length Record.

The Ethernet frame begins immediately after the pad byte so that the layer 3 [IP] header is 32Bit-aligned.

6.2 Timestamps

Description

The ERF format incorporates a hardware generated timestamp of the packet's arrival.

The format of this timestamp is a single little-endian 64-bit fixed point number, representing seconds since midnight on the first of January 1970.

The high 32-bits contain the integer number of seconds, while the lower 32-bits contain the binary fraction of the second. This allows an ultimate resolution of 2^{-32} seconds, or approximately 233 picoseconds.

Another advantage of the ERF timestamp format is that a difference between two timestamps can be found with a single 64-bit subtraction. It is not necessary to check for overflows between the two halves of the structure as is needed when comparing Unix time structures, which are also available to Windows user in the Winsock library.

Different DAG cards have different actual resolutions. This is accommodated by the lowermost bits that are not active being set to zero. In this way the interpretation of the timestamp does not need to change when higher resolution clock hardware is available.

Continued on next page

6.2 Timestamps, continued

Description,continued

Example code

Here is some example code showing how a 64-bit ERF timestamp (erfts) can be converted into a struct timeval representation (tv).

```
unsigned long long lts;
struct timeval tv;

 lts = erfts;
 tv.tv_sec = lts >> 32;
 lts = ((lts & 0xffffffffULL) * 1000 * 1000);
 lts += (lts & 0x80000000ULL) << 1;      /* rounding */
 tv.tv_usec = lts >> 32;
 if(tv.tv_usec >= 1000000) {
    tv.tv_usec -= 1000000;
    tv.tv_sec += 1;
      }
```