DAG 4.5G2/G4/GF Card
User Guide

EDM01-18

www.endace.com

Published by:

Endace Limited

Building 7

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

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

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

Protection Against Harmful Interference

When present on equipment this manual pertains to, 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.

Disclaimer

Whilst every effort has been made to ensure accuracy, neither Endace Limited nor any employee of the company, shall be liable on any ground whatsoever to any party in respect of decisions or actions they may make as a result of using this information.

Endace Limited has taken great effort to verify the accuracy of this manual, but assumes no responsibility for any technical inaccuracies or typographical errors.

In accordance with the Endace Limited policy of continuing development, design and specifications are subject to change without notice.

Table of Contents

Chapter 1: Introduction

Overview

Note: Unless specifically stated otherwise, throughout this User Guide "*DAG 4.5 card*" refers to the DAG 4.5G2/G4/and GF cards.

The Endace DAG 4.5 card provides the means to transfer data at the full speed of the network into the memory of the host PC, with zero packet loss guaranteed in even worst-case conditions. Further, unlike a NIC, Endace products actively manage the movement of network data into memory without consuming any of the host PC's resources. The full attention of the CPU remains focused on the analysis of incoming data without a constant stream of interruptions as new packets arrive from the network. For a busy network link, this feature has a turbo-charging effect similar to that of adding a second CPU to the system.

The DAG 4.5 card provides independent two or four port Ethernet network monitoring at Gigabit speeds and supports header only or variable length capture. It is capable of transmitting and receiving on each channel simultaneously allowing a single card to operate inline, monitoring and transmitting in both directions on a full duplex link.

Purpose of this User Guide

The purpose of this User Guide is to provide you with an understanding of the DAG card architecture and functionality and to guide you through the following:

- Installing the card and associated software and firmware,
- Configuring the card for your specific network requirements,
- Running a data capture session,
- Synchronizing clock time,
- Output data formats

You can also find additional information relating to functions and features of the DAG 4.5 card in the following documents which are available from the Support section of the Endace website at www.endace.com:

- EDM04-08 Configuration and Status API Programming Guide
- EDM04-10 Data Stream Manager API Programming Guide
- EDM04-07 DSM Loader User Manual

This User Guide and the Linux and Windows® Guides are also available in PDF format on the installation CD shipped with your DAG 4.5 card.

System Requirements

General

The minimum system requirements for the DAG 4.5 card are :

- PC, at least Intel Xeon 1.8GHz or faster,
- Minimum of 256 MB RAM,
- At least one free PCI-X slot supporting 66-133 MHz operation,
- Software distribution requires 60MB free space,

Operating System

This User Guide assumes you are installing the DAG card in a PC which already has an operating system installed.

However for convenience, a copy of Debian Linux 3.1 (Sarge) is provided as a bootable ISO image on the CD that is shipped with the DAG card.

To install either the Linux/FreeBSD or Windows® operating system please refer to the following documents which are also included on the CD that is shipped with the DAG card.

- EDM04-01 Linux FreeBSD Software Installation Guide
- EDM04-02 Windows® Software Installation Guide

Other Systems

For advice on using an operating system that is substantially different from either of those specified above, please contact Endace Customer Support at support@endace.com

Card Description

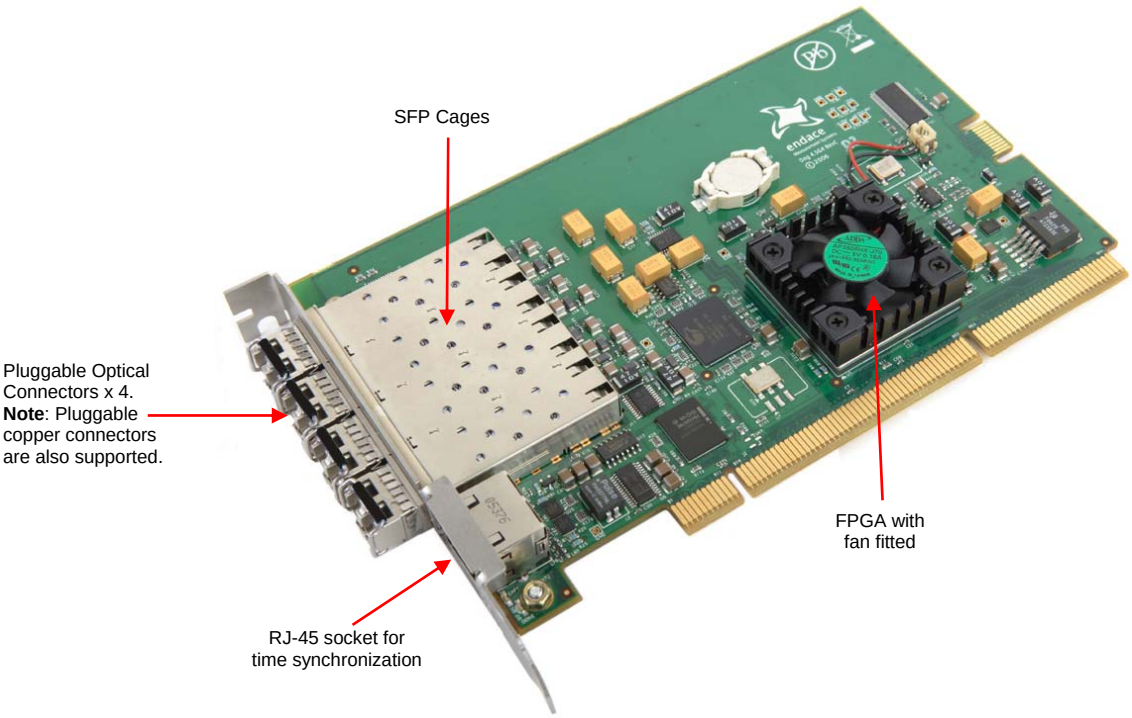
The DAG 4.5 card provides two or four optical or copper Gigabit Ethernet interfaces. It is capable of full line rate (1,000Mbps) capture and transmission of Ethernet traffic and is protocol independent. Full packet capture at line rate allows recording of all header information and/or payload with a high precision timestamp.

> **Note:** The DAG 4.5G2/GF has two ports, the DAG 4.5G4 has four ports.

The key features of the card are:

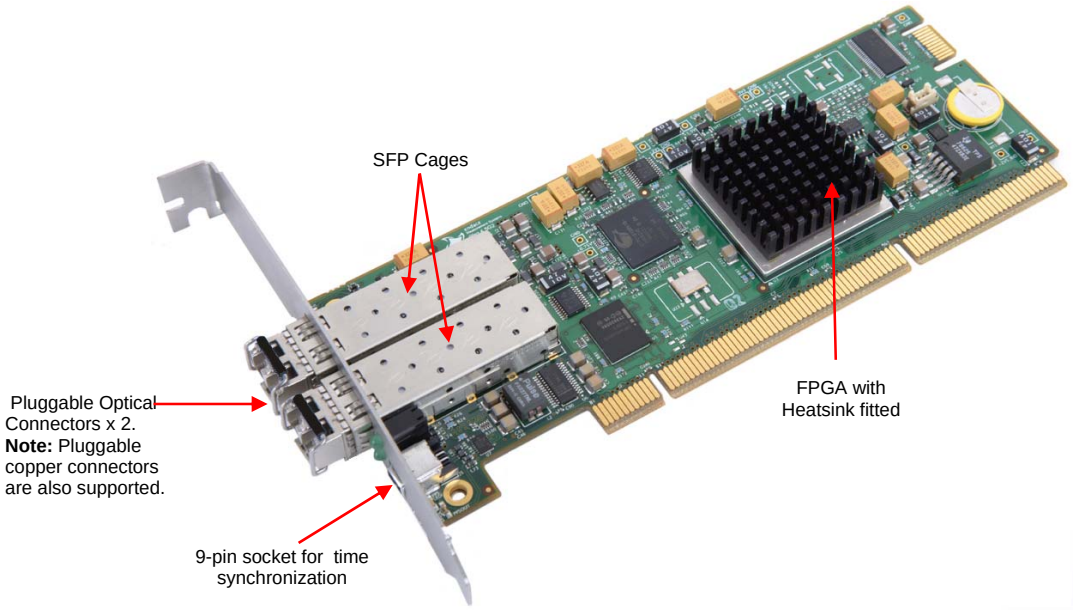
- Two or four SFP ports for 1000Base-SX or 1000Base-LX optical Ethernet or 1000Base-T copper Ethernet,
- Header only or variable length capture,
- Full line rate transmit,
- 100% capture into host memory at full line rate for IP packets from 48 to 9600 bytes
- Conditioned clock with PPS input and local synchronization capability.
- PCI-X 64-bit 66/100/133 MHz bus interface with 3V signaling.
- Failsafe optical relays to connect the two ports on the card in event of a power failure (DAG 4.5GF only).

The DAG 4.5G4 card is shown below:



> **Note:** Although the DAG 4.5G4 supports full line rate capture or transmit across all four ports, maximum combined throughput (i.e. simultaneous capture and transmission) is limited by the bandwidth of the PCI-X.

The DAG 4.5G2 card is shown below:



Note: The DAG 4.5G2/GF card is capable of simultaneous capture and transmission at full line rate across both ports.

Card Architecture

Serial Ethernet network data received by two 1000Base interfaces flows directly into the Field Programmable Gate Array (FPGA).

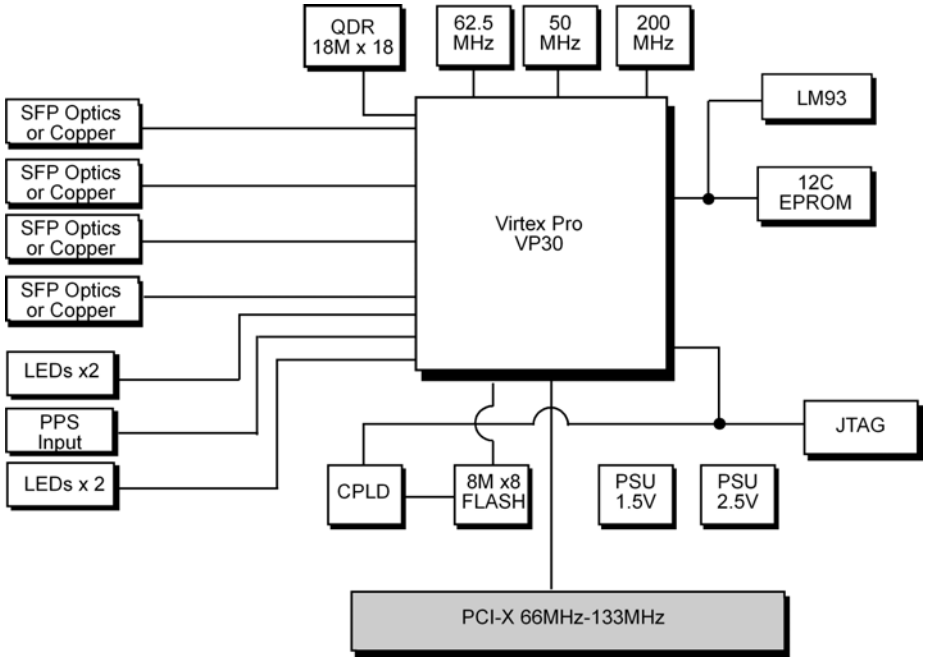
The FPGA contains the packet processor, PCI-X interface logic and the DAG Universal Clock Kit (DUCK) timestamp engine. The DUCK provides high resolution per-packet timestamps which can be accurately synchronized.

> **Note:** For further information on the DUCK and time synchronizing please refer to *Chapter 5: Synchronizing Clock Time* later in this User Guide.

Because of component close association, packets are time-stamped accurately. Time stamped packet records are stored in an external FIFO memory before transmission to the host.

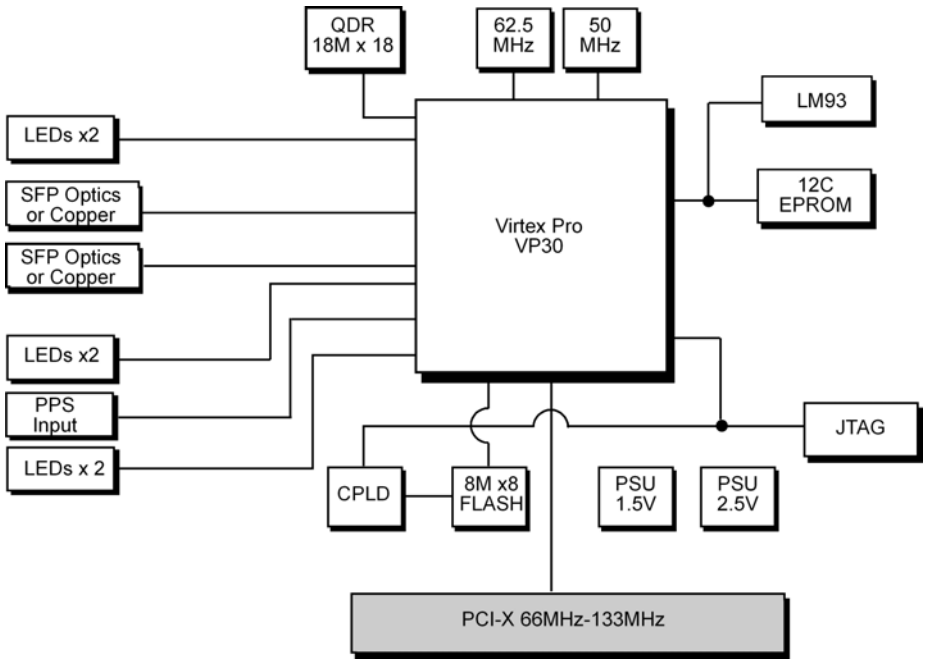
DAG 4.5G4

The diagram below shows the DAG 4.5G4 card's major components and flow of data.



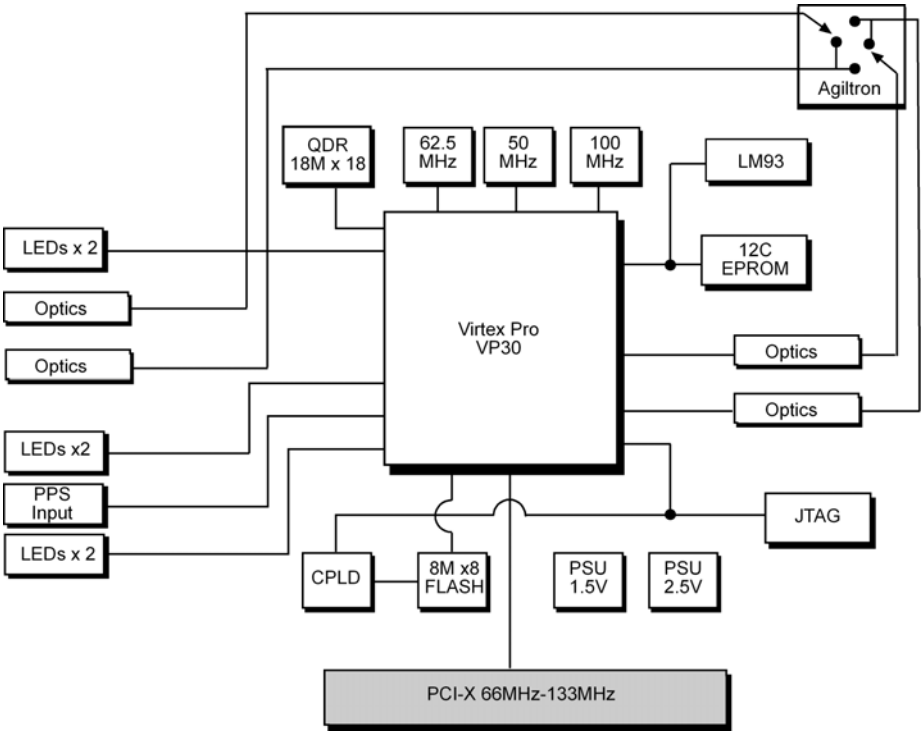
DAG 4.5G2

The diagram below shows the DAG 4.5G2 card's major components and flow of data.



DAG 4.5GF

The diagram below shows the DAG 4.5GF card's major components and flow of data:



Extended Functions

Failsafe Relays

The DAG 4.5GF has failsafe optical relays which allow you to connect the two ports in the event of a power interruption or software or hardware failure on the host. This means that in such an event the traffic can be switched to bypass the DAG card while still maintaining network connectivity.

The failsafe relays are controlled via an optical switch on the DAG card and can be configured to either fail-open or fail-closed.

> **Note:** This feature is <u>not</u> available on the DAG 4.5G2/G4 cards.

Fail-Open

Fail-open is the default configuration and is for use in situations where the nature of the event means that traffic must be stopped. In fail-open mode the network connection will switch to open circuit in the event of a failure which is effectively the same as disconnecting the DAG card from the network.

Fail-Closed

Fail-closed mode is for use in situations when the nature of the event means that traffic must still flow. In this mode the two ports are connected together so that all traffic received on Port A is transmitted on Port B and all traffic received on Port B is transmitted on Port A.

> **Note:** To implement the fail-closed mode you <u>must</u> engage the failsafe relays using the `dagwatchdog` command. For further information on using `dagwatchdog` please refer to *Starting a Session* in *Chapter 4: Capturing Data* later in this User Guide.

Extended Functions (cont.)

Data Stream Manager

The DAG 4.5 card supports the Data Stream Manager (DSM) feature. DSM allows you to drop or route packets to a particular receive stream based on the packet contents, physical port and the output of two load balancing algorithms.

The DSM logic is implemented in firmware on the DAG card, it does not require intervention from the host CPU once it is configured.

Filter / Load Balancing Block

Packets are received from the line and stamped with an ERF header, then passed to the filter and load balancing block. The filter block applies eight bit-mask filters simultaneously to the start of the packet, producing a single true/false value for each filter.

The load balancing block applies two algorithms to the packet data, also producing one true/false Boolean output per algorithm.

Lookup Table Block

The lookup table accepts the filter and load balancing outputs. It also receives the physical port the packet arrived on and calculates a classification (also known as color) for the packet.

Coloriser and Drop Block

The color is then passed to the Coloriser And Drop (CAD) block to check if the packet should be dropped. If not the color is inserted into the packet ERF record header and the packet record is passed to the packet record multiplexer.

Packet Record Multiplexer (ERF MUX)

The ERF MUX looks at the color information contained in the packet record and determines which receive stream the packet record should be routed to.

> **Note:** For detailed information on using the Data Stream Manager please refer to *EDM04-10 Data Stream Management API Programming Guide* and *EDM04-07 dsm-loader User Guide* available from the Support section of the Endace website at www.endace.com.

Inline Forwarding

The DAG 4.5 card supports inline forwarding which enables the card to receive and transmit packets directly from a single memory. This allows you to forward packets from one interface to the other without the requirement to copy them. Using inline forwarding you can receive, inspect, filter and forward packets between two ports.

The DAG 4.5 can perform inline forwarding at 100% of line rate in both directions. However when using all four ports on the DAG 4.5G4 the maximum throughput may be limited by the bandwidth of the PCI-X.

`dagfwddemo` which is a tool supplied with your DAG card demonstrates how you can apply a user-defined BSD Packet Filter (BPF) to the traffic forwarded by the DAG card. Packets which match the filter are forwarded, while packets that do not match are dropped.

For more detailed information on inline forwarding and using `dagfwddemo` please refer to the *EDM 04-04: dagfwddemo User Guide* available from the support section of the Endace website at www.endace.com.

Chapter 2: Installation

Introduction

The DAG 4.5 card can be installed in any free PCI-X slot. It operates at 66MHz, 100MHz or 133MHz PCI-X mode but will not operate in 32-bit or 64-bit PCI slots. Higher speed slots will achieve better performance.

Although the DAG driver supports up to sixteen DAG cards in one system, the DAG card makes heavy use of PCI-X bus data transfer resources. Therefore bandwidth limitations mean you should not have more than one card on a single PCI-X bus.

DAG Device Driver

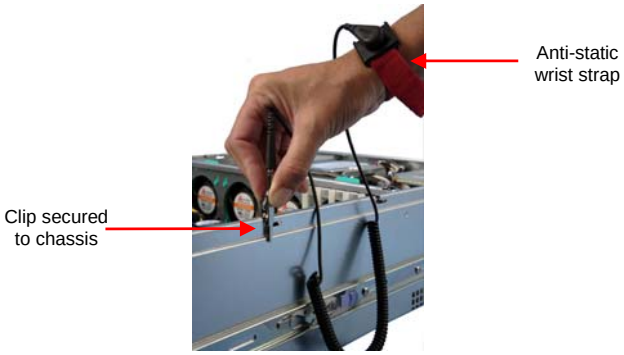
The DAG device driver must be installed before you install the DAG card.

If you have not already completed this please follow the instructions in *EDM04-01 Linux FreeBSD Software Installation Guide* or E*DM 04-02 Windows® Software Installation Guide* as appropriate, which are included on the CD shipped with the DAG card.

Inserting the DAG Card

Caution: It is very important to protect both the PC and the DAG card from damage by electro-static discharge (ESD). Failure to do so could cause damage to components and subsequently cause the card to partially or completely fail.

- Turn power to the computer OFF,
- Remove the PCI bus slot screw and cover,
- Using an approved ESD protection device attach the end with the strap to your wrist and pull or clip firmly so there is firm contact with your wrist,
- Securely attach the clip on the other end of the strap to a solid metal area on the PC chassis as shown below,



- Insert the DAG card into PCI bus slot ensuring it is firmly seated ,
- Check the free end of the card fits securely into the card-end bracket that supports the weight of the card,
- Secure the card with the bus slot cover screw,
- Turn power to the computer ON.

Port Connectors

The DAG 4.5G2/GF card has 2 x Small Form Factor (SFP) socket connectors. The DAG 4.5G4 card has 4 x SFP socket connectors. Each connector consists of an optical fiber or copper transmitter and receiver.

The upper connector of each pair is used for the transmit signal and the bottom connector of each pair is used for the received signal.

> **Note:** The DAG 4.5G.2/G4 supports both optical and copper transceivers. The DAG 4.5GF supports optical transceivers only.

The DAG 4.5G4 has an 8-pin socket located below the optical port connectors on the card bracket. The DAG 4.5G2/GF has a 4-pin socket in the same location. These sockets are available for connection to an external time synchronization source.

Caution: Never connect anything other than a PPS input to the time synchronization sockets.

Pluggable Optical Transceivers

Overview

The DAG 4.5G2/G4 card supports industry standard Small Form-factor Pluggable (SFP) optical transceivers.

> **Note:** The DAG 4.5GF transceivers are not pluggable

The transceivers consists of two parts:

- Mechanical chassis attached to the circuit board
- Transceiver unit which may be inserted into the chassis

> **Note:** You must select the correct transceiver type to match the optical parameters of the network to which you want to connect. Configuring the card with the wrong transceiver type may damage the card.

You can connect the transceiver to the network via LC-style optical connectors.

For further information on pluggable optical transceivers please refer to the Endace website at www.endace.com/dagpluggable.htm.

Setting Power

The optical power range depends on the particular SFP module that is fitted to the DAG card. However Endace recommends you use the 1000baseSX 850nm short range modules which are shipped with the card.

Optical power is measured in dBm. This is decibels relative to 1 mW where 10 dB is equivalent to a factor of 10 in power. A negative optical power value indicates power that is less than 1 mW. The most sensitive devices can work at power levels down as low as –30dBm or 1µW.

The DAG 4.5 card optics power module specification is shown below:

Manufacturer	Part #	Fiber	Data Rate	Max Pwr		Min Pwr	
Finisar	FTLF8519P2	MMF	1Gbps	Rx	0dBm	Rx	-20dBm
				Tx	-3dBm	Tx	-9dBM
Finisar	FTRJ1319P1	SMF	1Gbps	Rx	-9dBm	Rx	-22dBm
				Tx	-3dBm	Tx	-9.5dBm

Pluggable Optical Transceivers (cont)

Power Input

Note: The optical power input to the DAG card must be within the receiver's dynamic range of 0 to -22dBm. If it is slightly outside of this range it will cause an increased bit error rate. If it is significantly outside of this range the system will not be able to lock onto the signal.

When power is above the upper limit the optical receiver saturates and fails to function. When power is below the lower limit the bit error rate increases until the device is unable to obtain lock and fails. In extreme cases, excess power can damage the receiver.

When you set up the DAG card you should measure the optical power at the receiver and ensure that it is within the specified power range. If it is not, adjust the input power as follows:

- Insert an optical attenuator if power is too high, or
- Change the splitter ratio if power is too high or too low.

Splitter Losses

Splitters have the insertion losses either marked on their packaging or described in their accompanying documentation. General guidelines are:

- A 50:50 splitter will have an insertion loss of between 3 dB and 4 dB on each output
- 90:10 splitter will have losses of about 10 dB in the high loss output, and <2 dB in the low loss output

Note: A single mode fiber connected to a multi-mode input will have minimal extra loss. However a multi-mode fiber connected to a single mode input will create large and unpredictable loss.

Pluggable Copper Transceivers

The DAG 4.5G2/G4 card supports industry standard Small Form-factor pluggable (SFP) copper transceivers.

The transceivers consists of two parts:

- Mechanical chassis attached to the circuit board
- Transceiver unit which may be inserted into the chassis

Endace recommends that you use Cat6 copper cable. The DAG 4.5 card copper module specification is shown below:

Part	Type	Data Rate
FCMJ8521C	10/100/1000 Base-T	>1.25Gbps

Chapter 3: Configuring the Card

Introduction

Configuring the DAG card for data capture involves the following steps:

- Loading the images and programming the FPGA,
- Setting the link,
- Checking the link,
- Configuring the connections,
- Capturing data.

The `dagconfig` tool which is also supplied with the DAG 4.5 card allows you to configure the card to your specific network requirements as well as view interface statistics.

Sample `dagconfig` outputs are shown later in this chapter.

Line Types

Overview

It is important that you understand the physical characteristics of the network to which you want to connect before you begin configuring the card.

There are various Ethernet line speeds and corresponding protocols which are identified using the IEEE naming convention. Each line speed has a set of requirements associated with it relating to the type of cable, maximum allowable distance, etc.

> **Note:** If you are unsure about which of the options listed below apply to your network, please contact your Network Administrator for further information.

Supported Line Types

The line characteristics supported by the DAG 4.5 card are described below.

Type	Description
10Base-T	10 Mbps over two pairs of twisted telephone cable.
100Base-TX	100 Mbps over two pairs of shielded or unshielded twisted Cat 5 copper cable.
1000Base-T	1000Mbps over four pairs of balanced Cat5 or Cat6 copper cable.
1000Base-LX	1000Mbps over single mode or multi mode fiber optic cable with long wavelength laser driver (1310nm)
1000Base-SX	1000Mbps over single mode or multi mode fiber optic cable with short wavelength laser driver (850nm)

> **Note:** For more detailed information regarding Ethernet line types and speeds, please refer to IEEE Standard 802.3 available from the IEEE website at www.ieee.org.

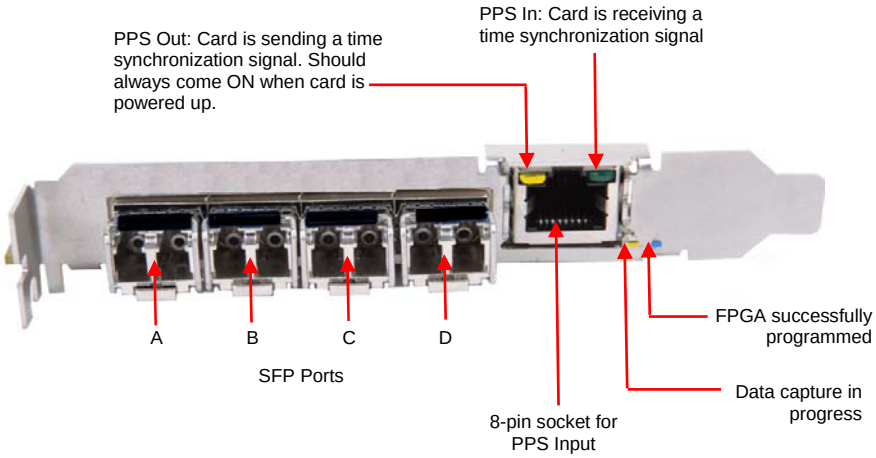
LEDs and Inputs

Before you begin to configure the DAG card it is important to understand the function of the various LEDs associated with the card, as well as the sockets on the PCI bracket.

> **Note:** The LEDs and sockets on the DAG 4.5G4 card are different to those on the DAG 4.5G2/GF card.

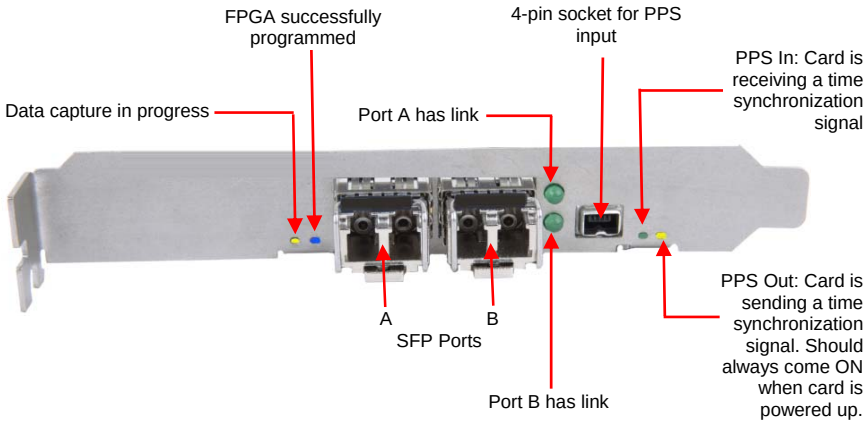
DAG 4.5G4

The LEDs and sockets on the DAG 4.5G4 bracket are shown below:



DAG 4.5G2

The LEDs and sockets on the DAG 4.5G2 bracket are shown below:



Receiver Port Signal Levels

The card receiver ports are the lower of each dual-LC-style connector, the closest to the PCI slot.

The DAG card supports 1310nm Singlemode Fiber (SMF) attachments with optical signal strength between -3dBm and -20dBm , and 850nm Multimode Fiber (MMF) attachments with optical signal strength between 0dBm and -22dBm. If there is doubt check the light levels on the card receiver port are correct using an optical power meter.

> **Note**: If not in use you should cover the card transmit ports with LC-style plugs to prevent dust and mechanical hazards damaging the optics.

Load the FPGA Image

Before you can configure the card for capture you must load the card with the appropriate FPGA image:

- For the **DAG 4.5G2/GF** card use:

```
dagrom –d0 -rvp -f dag45g2pcix-terf-dsm.bit
```

- For the **DAG 4.5 G4** card use:

```
dagrom –d0 –rvp –f dag45g4pcix-terf-dsm.bit
```

Display Current Configuration

Once you have loaded the appropriate image you should run the `dagconfig` tool without arguments to display the current card configuration and verify the firmware has loaded correctly, using:

`dagconfig –d0` (where "`0`" is the device number of the DAG card)

An explanation of the `dagconfig` outputs is shown below:

DAG 4.5G2/GF Output

```
Firmware: edag45g2pci_terf-ntr_pci_v2_3 2vp30ff1152 2006/08/08
17:29:14 (user)
Serial : 0

MAC Address A : 00:00:00:00:00:00
MAC Address B : 00:00:00:00:00:00

Port A: nic noeql 1000 Mbps drop_count=0 enablea
Port B: nic noeql 1000 Mbps drop_count=0 enableb

GPP:
slen=1536 varlen

PCI Burst Manager:
133MHz buffer size = 128 rx_streams = 1 tx_streams = 1 nodrop

Memory Streams:
mem=112:16

TERF:
terf_strip32
```

Number of packets dropped during current capture session. resets to "0" if session is restarted.

Sets (eql) or unsets (noeql) equipment loopback. **Note:** eql loops back to the PCI bus.

DAG card will operate in auto-negotiation mode (nic) or not (nonic).

Enables or disables the port for capture.

First 1536 bytes of packet will be captured. **Note:** Snaplength value must be a multiple of 8 and in the range 48 to 9600.

Variable length (varlen) or fixed length (novarlen) packet capture.

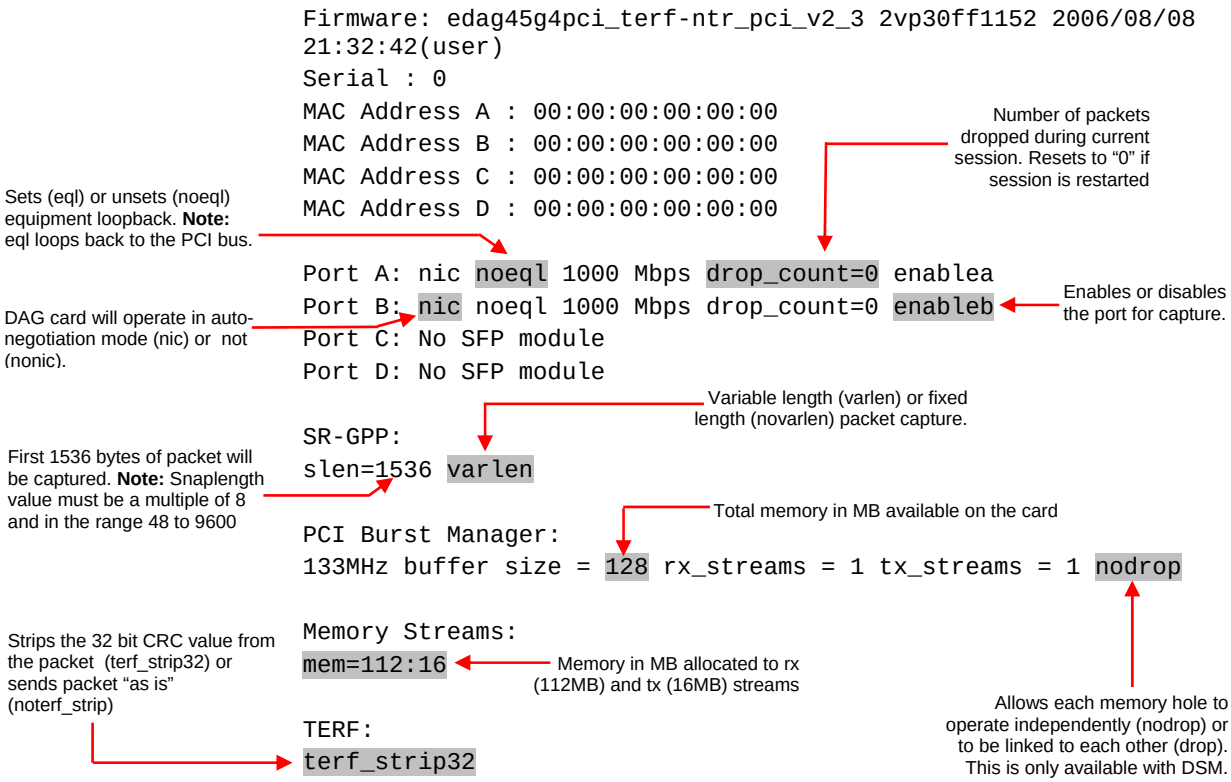
Total memory in MB available on the card.

Strips the 32 bit CRC value from the packet (terf_strip32) or sends packet "as is" (noterf_strip)

Memory in MB allocated to receive (112MB) and transmit (16MB) streams.

Allows each memory hole to operate independently. This is only available with DSM.

DAG 4.5G4 Output



Note: If you wish to use the Data Stream Manager (DSM) feature you will need to revert to normal capture mode using the following command line:

```
dsm_loader –b
```

This will allow you to bypass DSM mode, and perform normal capture functions.

Auto-Negotiation

The DAG 4.5G2/G4/GF can operate in either "nic" or "nonic" mode In nic mode you must connect the DAG card directly to a Gigabit Ethernet switch or card with a full-duplex cable, in which case the DAG card will perform Gigabit Ethernet auto-negotiation.

The nonic mode is intended for use with optical fiber splitters. In this mode you must connect the receive socket of the DAG port to the output of an optical splitter inserted into a network link between two other devices. The transmit socket of the DAG should be unconnected.

In nonic mode, Gigabit Ethernet auto-negotiation is not performed. This allows you to use one splitter on each DAG receive port to monitor each direction of a full-duplex Gigabit Ethernet link.

Interface Statistics

When you have configured the card according to your specific requirements you can view the interface statistics to check the status of each of the links using:

```
dagconfig -d0 -si
```
← Display statistics once (-s) or at 1 sec intervals (-si)

Example outputs are shown below:

> **Note:** "1" indicates the condition is present on the link "0" indicates the condition is not present on the link. See *Status Conditions* later in this chapter for a full description of each of the status conditions.

No Link

The example below shows no link on either port A or port B

```
Port  Link  PLink  RFault  LOF  LOS
   A    0     0      0       1    1
   B    0     0      0       1    1
   A    0     0      0       1    1
   B    0     0      0       1    1
```

Link

The example below shows a valid link on both port A and port B

```
Port  Link  PLink  RFault  LOF  LOS
  A     1     1      0      0    0
  B     1     1      0      0    0
  A     1     1      0      0    0
  B     1     1      0      0    0
```

Status Conditions

A definition of each of the status conditions is described below:

Condition	Definition
LOS	Loss of Signal There is either no signal at the receiver or the optical signal strength is too low for the card to recognize.
LOF	Loss of Framing. Indicates Out of Frame (OOF) condition has been asserted for more than 2 milliseconds.
RFault	There is an error at the remote end of the link
Plink	The peer link is valid
Link	The link is fully validated.

Chapter 4: Capturing Data

Starting a Session

For a typical data capture session follow the steps listed below:

- Move to the directory in which you installed the DAG software
- Load the appropriate driver,
- Then load the appropriate FPGA image for the DAG card as described in *Load the FPGA Image* in *Chapter 3: Configuring the Card* earlier in this User Guide.
- Configure the card's physical layer and check the integrity of the physical layer to each DAG card. For example:

```
dagconfig –d0 default
```

- If you are configuring a DAG 4.5GF card and you wish to engage the failsafe relays, use:

```
dagwatchdog –d0 –p
```

  Note: For more information on the Failsafe Relay feature please refer to *Failsafe Relays* in *Chapter 1: Introduction* earlier in this user guide.

- If you wish to use the Data Stream Manager (DSM) feature you will need to bypass DSM mode to enable you to perform normal capture. You can revert to normal capture mode using:

```
dsm_loader –b
```

- Start the capture session using:

```
dagsnap -d0 –v -o tracefile
```

  Note: You can use `-v` to provide user information during a capture session although you may want to omit it for automated trace runs.

By default `dagsnap` will run indefinitely but can be stopped using *CTRL+C*. You can also configure `dagsnap` to run for a fixed time period then exit.

Setting Captured Packet Size

Snaplength

Before you begin to capture data you can set the size that you want the captured packets to be. You can do this using the `dagconfig` tool to define the packet snaplength (`slen`).

> **Note:** The snaplength value must be a multiple of 8 and in the range 48 to 9600 inclusive.

By default `slen` which is the portion of the packet that you want to capture, is set to 1536. This means that only the first 1536 bytes of each packet will be captured. If for example you want to capture only the IP header of each packet you may want to set the length to a smaller value. Alternatively if you want to ensure you capture the whole packet you can set the length to a larger value.

> **Note:** The ERF header is not included in the `slen` value. Therefore a `slen` of 48 will produce a capture record of 48 bytes plus the number of bytes in the ERF header.

Setting Captured Packet Size (cont.)

Variable/Fixed Length

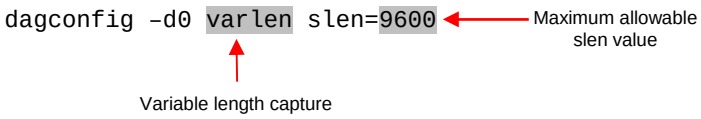
The DAG 4.5 card is able to capture packets in two ways. They are:

- Variable length capture (`varlen`)
- Fixed length capture (`novarlen`)

In variable length (`varlen`) mode the card will capture the whole packet, providing its size is less than the `slen` value. Therefore to use this capture mode effectively you should set the `slen` value to the largest number of bytes that a captured packet is likely to contain.

Any packet that is larger than the `slen` value will be truncated to that size. Any packet that is smaller than the `slen` value will be captured at its actual size therefore producing a shorter record which save bandwidth and storage space.

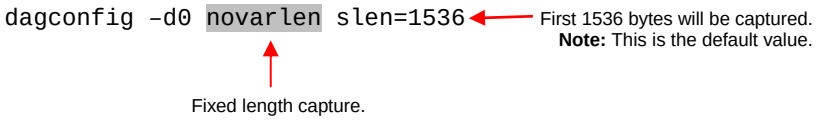
The example below shows a configuration for variable length full packet capture:



In fixed length (`novarlen`) mode the card will capture all packets at the same length. Any packet that is longer than the `slen` value will be truncated to that size, in the same way as for `varlen` capture. However any packet that is shorter than the `slen` value will be captured at its full size and then padded out to the size of the `slen` value.

This means that in `novarlen` mode you should avoid large `slen` values because short packets arriving will produce records with a large amount of padding which wastes bandwidth and storage space.

The example below shows a configuration for fixed length packet capture that will produce a 64-byte record:



Note: For Ethernet records a 64 byte record is made up of 46 bytes of payload and 18 bytes of ERF header. For more information on Ethernet records please refer to *Chapter 6: Data Formats* later in this user guide.

High Load Performance

Overview

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

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. If not the buffer will eventually fill and packet records will be lost.

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.

In Linux and Free BSD, when the PC buffer fills, the following message displays on the PC screen:

```
kernel: dagN: pbm safety net reached 0xNNNNNNNN
```

The same message is also printed to log /var/log/messages. In addition, when the PC buffer fills the "Data Capture" LED on the card will flash or flicker, or may go OFF completely.

In Windows® no screen message displays to indicate when the buffer is full. Please contact Endace Customer Support at support@endace.com for further information on detecting buffer overflow and packet loss in Windows®

Detecting Packet Losses

Once the buffer fills, any new packets arriving will be discarded by the DAG card until some data is read out of the buffer to create free space.

You can detect any such losses by observing the Loss Counter (`lctr` field) of the Extensible Record Format [ERF]. See *Chapter 6: Data Formats* later in this User Guide for more information on the Endace ERF.

Transmitting

Configuration

The DAG 4.5 card is able to transmit as well as receive packets and can capture received traffic while transmitting. This allows you to use capture tools such as `dagsnap`, `dagconvert`, and `dagbits` while `dagflood` is sending packets.

To configure the DAG card for transmission, you must allocate some memory to a transmit stream. By default, 16 MB of memory is allocated to the tx stream and the remainder is allocated to the rx stream.

In the `dagconfig` output, `buffer size =nMB` indicates that a total of `n` MBs of memory have been allocated to the DAG card in total. You can split this allocation between the receive and transmit stream buffers to suit your own requirements. The split is displayed as a ratio as shown below:

> `mem=X:Y`, where
>
> `X` is the memory allocated in MB to the rx stream
> `Y` is the memory allocated in MB to the tx stream.

For instance you can split 128MB of memory evenly between the tx and rx streams using:

```
dagconfig –d0 mem=64:64
```

Note: You can not change the stream memory allocations while packet capture or transmission is in progress.

Explicit Packet Transmission

The operating system does not recognize the DAG card as a network interface and will not respond to ARP, ping, or router discovery protocols.

The DAG card will only transmit packets that are explicitly provided by the user. This allows you to use the DAG card as a simple traffic load generator.

You can also use it to retransmit previously recorded packet traces. The packet trace is transmitted at 100% line rate. The packet timing of the original trace file is not reproduced.

Trace Files

You can use the `dagflood` tool to transmit ERF format trace files, providing the files contain only ERF records of the type matching the current link configuration.

If you do not have any ERF trace files available, you can use `daggen` to generate trace files containing simple traffic patterns. This allows the DAG card to be used as a test traffic generator.

For further information on using `daggen` please refer to the *EDM04-06: Daggen User Guide* available from the Support section of the Endace website at www.endace.com.

Increasing Buffer Size

You can increase the size of the host PC buffer to enable it to cope 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 Linux/BSD 128MB or more is recommended. However you can change the amount of reserved memory by editing the file `/etc/modules` as follows

```
# 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
```

For Windows® the upper limit is 32MB. This is usually sufficient however if you do need to increase the amount of reserved memory please contact Endace customer support at support@endace.com for more information

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

> **Note:** In a 32-bit system the value of dsize multiplied by the number of DAG cards in the system must be less than the amount of physical memory installed, as well as less than 890MB.

Chapter 5: Synchronizing Clock Time

Overview

The Endace DAG cards have sophisticated time synchronization capabilities, which allow for high quality timestamps, optionally synchronized to an external time standard.

The core of the DAG synchronization capability is known as the DAG Universal Clock Kit (DUCK).

An independent clock in each DAG card runs from the PC clock. The card's clock is initialized 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 designed to reduce time variance between sets of DAG cards or between DAG cards and coordinated universal time [UTC].

You can obtain an accurate time reference by connecting an external clock to the DAG card using the time synchronization connector. Alternatively you can use the host PCs clock in software as a reference source without any additional hardware.

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

Common Synchronization

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

Common synchronization sources include GPS or CDMA (cellular telephone) time receivers.

Endace also provides the TDS 2 Time Distribution Server modules and the TDS 6 units that enable you to connect multiple DAG cards to a single GPS or CDMA unit.

For more information please refer to the Endace website at http://www.endace.com/accessories.htm , or the *TDS 2/TDS 6 Installation Manual*.

Timestamps

ERF files contains a hardware generated timestamp of each packet's arrival. The format of this timestamp is a single little-endian 64-bit fixed point number, representing the number of seconds since midnight on the 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.

The ERF timestamp allows you to find the difference between two timestamps using a single 64-bit subtraction. You do not need to check for overflows between the two halves of the structure as you would need to do when comparing Unix time structures.

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.

Example

Below is 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;
      }
```

Configuration Tools

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

Synchronization settings are controlled by the `dagclock` utility.

Note: You should only run `dagclock` after you have loaded the appropriate Xilinx images. If at any stage you reload the Xilinx images you must rerun `dagclock` to restore the configuration.

```
 dagclock -h
Usage: dagclock [-hvVxk] [-d] [-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	the DAG device
-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

Note: By default, all DAG cards listen for synchronization signals on their RS-422 port, and do not output any signal to that port

```
dagclock –d0
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
```

Card with Reference

Overview

To obtain the best timestamp accuracy you should connect the DAG card to an external clock reference, such as a GPS or CDMA time receiver.

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 initialize the DUCK.

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

Pulse Signal from External Source

The DAG time synchronization connector supports an RS-422 (PPS) signal from an external source. This is derived directly from an external reference source, or distributed through the Endace TDS 2 (Time Distribution Server) module which allows two DAG cards to use a single receiver. It is also possible for more than two cards to use a single receiver by "daisy-chaining" TDS-6 expansion modules to the TDS-2 module. Each TDS-6 , module provides outputs for an additional 6 DAG cards.

Synchronize to an external source as follows:

```
dagclock –d0
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 the Time Distribution Server

You can connect the TDS 2 module to the DAG card using standard RJ-45 Ethernet cable including existing RJ-45 building cabling. The TDS may be located up to 600m (2000ft) from the DAG card depending upon the quality of the cable used, possible interference sources and other environmental factors. Please refer to the *TDS2/TDS6 User Guide* for more in formation

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

Testing the Signal

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

To test the signal is being received correctly and has the correct polarity use the `dagpps` tool as follows:

```
dagpps –d0
```

`dagpps` measures the input state many times over several seconds, displaying the polarity and length of input pulse.

Single Card No Reference

When a single card is used with no external reference, the card can be synchronized to the host PC clock. Most PC clocks are not very accurate by themselves, 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 not as smooth because the PC clock is adjusted in small jumps. However the DUCK clock does not drift away from UTC.

The synchronization achieved with this method is not as accurate as using an external reference source such as GPS.

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

```
dagclock –d0 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
```

Two Cards No Reference

Overview

If you are using two DAG cards in a single host PC with no reference clock, you must synchronize the cards using the same method if you wish to compare the timestamps between the two cards. You may wish to do this for example if the two cards monitor different directions of a single full-duplex link. You can synchronize the cards in two ways:

- One card can be a clock master for the second. This is useful if you want both cards to be accurately synchronized with each other, but not so for absolute time of packet time-stamps, or
- One card can synchronize to the host and also act as a master for the second card

Two Cards No Reference (cont.)

Synchronizing with Each Other

Although the master card's clock will drift against UTC, the cards will still be locked together. This is achieved by connecting the time synchronization connectors of both cards using a standard RJ-45 Ethernet cross-over cable.

Configure one of the cards as the master so that the other defaults to being a slave as follows:

```
dagclock –d0 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
```

> **Note:** The slave card configuration is not shown as the default configuration will work.

Synchronizing with Host

To prevent the DAG card clock 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.

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

```
dagclock –d0 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.

Connector Pin-outs

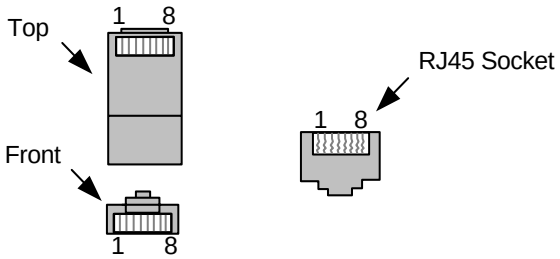
Overview

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 and plugs and sockets are shown below:

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



Normally you should connect the GPS input to the A channel input (pins 3 and 6).

The DAG card can also output a synchronization pulse for use when synchronizing two DAG cards without a GPS input. The synchronization pulse is output on the Out A channel (pins 1 and 2).

Ethernet Crossover Table

You can use a standard Ethernet crossover cable to connect the two cards as shown below:

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-

Chapter 6: Data Formats

Overview

DAG Cards produce trace files in their own native format called ERF (Extensible Record Format). The ERF type depends upon the type of connection you are using to capture data.

The DAG 4.5 supports the following ERF Types:

ERF Type	Description
2	TYPE_ETH Ethernet Variable Length Record
16	TYPE_DSM_COLOR_ETH Ethernet Variable Length Record

The ERF file contains a series of ERF records with each record describing one packet. An ERF file consists only of ERF records, there is no special file header which allows concatenation and splitting to be performed arbitrarily on ERF record boundaries.

Generic Header

All ERF records share some common fields. 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 or-ordering is applied.

The generic ERF header is shown below:

<table>
<tr><th>Byte 3</th><th>Byte 2</th><th>Byte 1</th><th>Byte 0</th></tr>
<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/color</td><td colspan="2">wlen</td></tr>
<tr><td colspan="4">(rlen - 16) bytes of record</td></tr>
</table>

Generic Header (cont.)

The fields are described below:

timestamp The time of arrival of the cell, an ERF 64-bit timestamp. See *Timestamps* in *Chapter 5: Synchronizing Clock Time* earlier in this User Guide for more information.

type One of the following:
2: TYPE_ETH
16: TYPE_DSM_COLOR_ETH

flags This byte is divided into several fields as follows:

1-0: Enumerates capture interface 0-3
2: Varying record lengths
3: Truncated record (insufficient buffer space)
4: RX error (link layer error)
5: DS error (internal error)
6: Reserved
7: Reserved.

rlen Record length. Total length of the record transferred over the PCI bus to storage.

lctr Depending upon the ERF type this 16 bit field is either a loss counter of color field. The loss counter records the number of packets lost between the DAG card and the memory hole due to overloading on the PCI bus.

wlen Wire length. Packet length including some protocol overhead. The exact interpretation of this quantity depends on physical medium.

Note: The Ethernet frame begins immediately after the pad byte so that the layer 3 (IP) header is 32-bit aligned.

Type-2 Record

The TYPE_ETH variable length record is shown below:

<table>
<tr><th>Byte 3</th><th>Byte 2</th><th>Byte 1</th><th>Byte 0</th></tr>
<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" rowspan="2"></td></tr>
<tr><td colspan="2">(rlen - 18) bytes of record</td></tr>
</table>

Type 16 Record

The TYPE_DSM_COLOR_ETH record is shown below:

Byte 3	Byte 2	Byte 1	Byte 0
timestamp			
timestamp			
type 16	flags	rlen	
color		wlen	
offset	pad		
(rlen - 18) bytes of record			

The TYPE_16 color field is divided into several bit fields as shown below:

Bits	Attribute
0-5	Receive stream 0-63
6-13	Filter match bits (bit 6 = filter 0, bit 7 = filter 1 etc)
14	hlb0 (CRC calculation) output bit
15	hlb1 (parity calculation) output bit

Note: Because both the Type-2 and Type-16 Ethernet record header occupies 18 bytes instead of the standard 16 bytes, any payload captured will always be 2 bytes less than the slen value ie. a slen of 48 which will produce a 64 bit record made up of 18 bytes of header and 46 bytes of payload.

Because the slen value must always be a multiple of 8, this means that if you wish to capture 48 bytes of payload for instance, the slen must be to set to 56.

Chapter 7 Troubleshooting

Reporting Problems

If you have problems with a DAG card or Endace supplied software which you are unable to resolve, please contact Endace Customer Support at support@endace.com.

Supplying as much information as possible enables Endace Customer Support to be more effective in their response to you. The exact information available to you for troubleshooting and analysis may be limited by nature of the problem. However the following items will assist a quick resolution:

- DAG card[s] model and serial number.
- Host PC type and configuration.
- Host PC operating system version
- DAG software version package in use
- Any compiler errors or warnings when building DAG driver or tools
- 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`.
- Output of daginf
- Firmware versions from `dagrom -x`.
- Physical layer status reported by: `dagconfig`
- Link statistics reported by: `dagconfig -si`
- Extended statistics reported by: `dagconfig -ei`
- Network link configuration from the router where available.
- Contents of any scripts in use.
- 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.
- A small section of captured packet trace illustrating the problem.

Version History

The version history for this user guide is shown below.

Version	Date	Reason
1-2		
3	April 2006	DAG 4.5GF information added. Changes to formatting and layout.
4	August 2006	DAG 4.5GF information expanded Separate FPGA images for DAG 4.5G2/GF and DAG 4.5G4 Standardization of features with other similar DAG cards. Layout and diagrammatic changes.