



**endace**  
a c c e l e r a t e d

**DAG 4.3S Card  
User Guide**  
EDM01-10

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](mailto:support@endace.com)

[www.endace.com](http://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

**Copyright 2005 ©All rights reserved.** No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of the publisher.

**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 Measurement Systems 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 Measurement Systems 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 Measurement Systems policy of continuing development, design and specifications are subject to change without notice.

# Table of Contents

<b>Chapter 1: Overview</b>	<b>1</b>
1.1 User Manual Purpose	1
1.2 DAG 4.3S Card Product Description	2
1.4 DAG Card Extended Functions	3
1.5 EMI Suppression Ferrite Fitting	4
1.6 DAG 4.3S Card System Requirements	4
<b>Chapter 2: Installation</b>	<b>6</b>
2.1 Installation of Operating System and Endace Software	6
2.2 Insert DAG 4.3S Card into PC	6
2.3 DAG 4.3S Card Port Connector	7
2.4 Pluggable Optical Transceivers	7
<b>Chapter 3: Setting Optical Power</b>	<b>9</b>
3.1 DAG 4.3S Card Optical Power Input	9
3.2 Splitter Losses	10
<b>Chapter 4 Confidence Testing</b>	<b>12</b>
4.1 Interpreting DAG 4.3S Card LED Status	12
4.2 DAG 4.3S Card Capture Session	14
4.3 Configuration in WYSYCC Style	16
4.4 DAG 4.3S Card Configuration Options	16
4.5 Inspect Link Data	21
4.6 Reporting Problems	22
<b>Chapter 5: Running Data Capture Software</b>	<b>25</b>
5.1 Starting Capture Session	25
5.2 High Load Performance	27
5.3.1 DAG Packet Transmission	29
5.3.2 Inline Forwarding	32
<b>Chapter 6: Synchronizing Clock Time</b>	<b>34</b>
6.1 Configuration Tool Usage	35
6.2 Time Synchronization Configurations	35
6.2.1 Single Card no Reference Time Synchronization	36
6.2.2 Two Cards no Reference Time Synchronization	36
6.2.3 Card with Reference Time Synchronization	37
6.3 Synchronization Connector Pin-outs	38
<b>Chapter 7: Data Formats</b>	<b>41</b>
7.2 Timestamps	43



# Chapter 1: Overview

**Introduction** The installation of the Endace DAG 4.3S 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.

**Viewing this document** This document, DAG 4.3S Card User Manual is available on the installation CD.

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

- User Manual Purpose
- DAG 4.3S Card Product Description
- DAG 4.3S Card Architecture
- DAG Card Extended Functions
- EMI Suppression Ferrite Fitting
- DAG 4.3S Card System Requirements

## 1.1 User Manual Purpose

**Description** The purpose of this DAG 4.3S Card User Manual is to describe:

DAG 4.3S Card Product Description  
 DAG 4.3S Card Architecture  
 DAG Card Extended Functions  
 EMI Suppression Ferrite Fitting  
 DAG 4.3S Card System Requirements

**Prerequisite** 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 4.3S Card Product Description

**Description** The DAG 4.3S card is designed only for PCI-X buses. It is capable of cell and packet capture and generation on IP networks.



Figure 1-1. DAG 4.3S series PCI-X Card.

The DAG 4.3S card collects packet header and payload from ATM, POS, and supports ATM, PPP HDLC, CISCO, HDLC over OC40c SONET or SDH networks.

Full packet or cell capture at line rate allows recording of all header information payload and with a high precision timestamp, subject to bus and processor availability.

The DAG 4.3S is capable of transmitting packets at 100% line rate while simultaneously receiving packets at 100% line rate.

## 1.3 DAG 4.3S Card Architecture

**Description** Serial SONET optical network data received by optical interface flows into a physical layer ASIC, then immediately into the Field-Programmable Gate Array [FPGA].

The FPGA contains an Endace DAG Universal Clock Kit [DUCK] timestamp engine, packet record processor, and PCI-X interface logic.

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

## 1.3 DAG 4.3S Card Architecture, continued

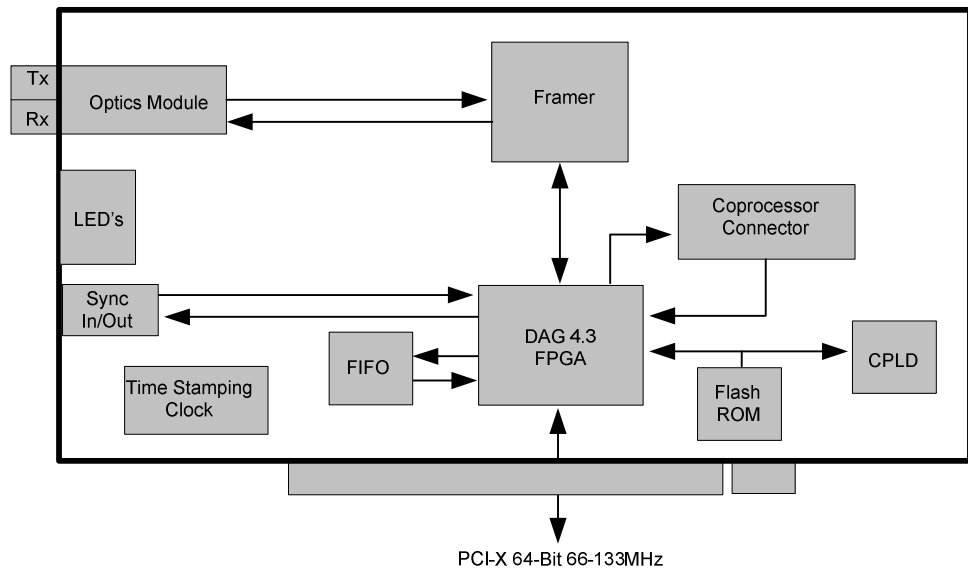


Figure 1-2. DAG 4.3S Card Major Components and Data Flow.

## 1.4 DAG Card Extended Functions

**Description** The DAG 4.3S functionality can be extended in many ways.

The co-processor allows for advanced features such as IP packet filtering in hardware at line rate. Contact the Endace customer support team at [support@endace.com](mailto:support@endace.com) to enable effective use of extended functions.



## 1.5 EMI Suppression Ferrite Fitting

**Description** In order to ensure compliance to EN55022 and FCC 47 Part 15 Class A Radiated emission limits, the supplied EMI Suppression Ferrite must be fitted to any cable connected to the Time Synchronisation Connector.

The Ferrite component supplied in the DAG 4.3S product packaging box must be fitted within 1cm of the plug housing..

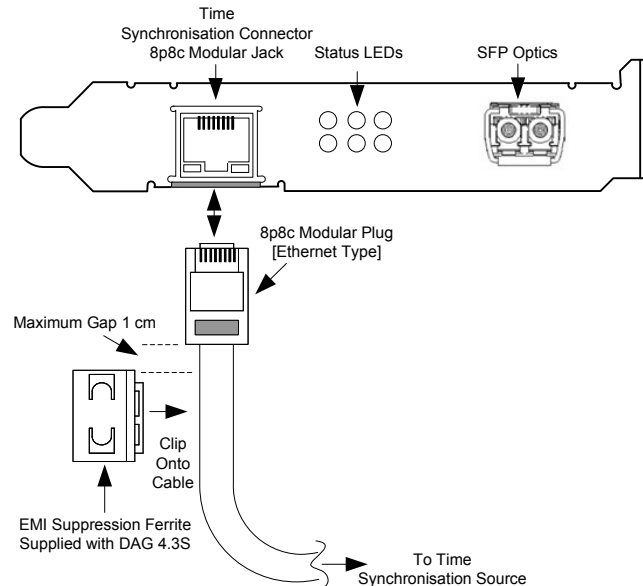


Figure 1-3. DAG 4.3S Card Ferrite Component.

## 1.6 DAG 4.3S Card System Requirements

**Description** The DAG 4.3S card and associated data capture system minimum operating requirements are:

- PC, at least Intel Xeon 1.8GHz or faster
- Intel E7500, ServerWorks Grand Champion LE/HE or newer chip set  
256 MB RAM
- At least one free PCI-X 1.0 slot supporting 66-133MHz operation
- Software distribution free space of 30MB
- Endace Linux Install CD requires 6 GB

PCI buses are not compatible with the DAG 4.3S card. Only PCI-X buses are supported.

**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.

## 1.3 DAG 4.3S Card Architecture, continued

**Different system** For advice on using a system substantially different from that specified above, contact Endace support at [support@endace.com](mailto:support@endace.com)

## Chapter 2: Installation

**Introduction** The DAG 4.3S card can be installed in any free PCI-X 1.0 slot. It will operate in 66, 100, and 133MHz PCI-X mode, however it will not operate correctly in 32 or 64-bit PCI slots.

Higher speed slots are recommended for best performance.

The DAG 4.3S card should be the only device on the PCI-X bus if possible as the card makes heavy use of PCI-X bus data transfer resources.

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

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

- Installation of Operating System and Endace Software
- Insert DAG 4.3S Card into PC
- DAG 4.3S Card Port Connector

### 2.1 Installation of Operating System and Endace Software

**Description** If the DAG device driver is not installed, before proceeding with the next chapter, install the software on Linux/FreeBSD operating systems by following the instructions in EDM04-01 Linux/FreeBSD Installation Guide.

To install the software on a Windows operating system, follow the instructions in EDM04-02 Windows Installation Guide.

Go to the next chapter of information when the DAG device driver is installed.

### 2.2 Insert DAG 4.3S Card into PC

**Description** Inserting the DAG 4.3S card into a PC involves accessing the bus slot, fitting the card, and replacing bus slot screw.

**Procedure** Follow these steps to insert the DAG 4.3S card.

**Step 1. Access bus Slot**

Power computer down.

Remove PCI-X bus slot cover.

## 1.3 DAG 4.3S Card Architecture, continued

### Procedure (continued)

#### Step 2. Fit Card

Insert into PCI-X bus slot.

#### Step 3. Replace bus Slot Screw

Secure card with screw.

#### Step 4. Power up Computer

## 2.3 DAG 4.3S Card Port Connector

**Description** There is one duplex LC-type optical port connector. The bottom connector is for the received signal, one nearest PCI-X slot.

The top port is for transmission and remains unconnected if a splitter is used for passive monitoring. If this port is unused, the transceiver optics should be covered or plugged to prevent ingress of dust.

An 8-pin RJ45 socket is used for time synchronization. This socket should never be connected to an Ethernet network or telephone line.

## 2.4 Pluggable Optical Transceivers

**Description** Some newer versions of the DAG 4.3S cards are available with pluggable optics. To provide compatibility with the broadest possible range of optical parameters, Endace offers the industry standard Small Form-factor Pluggable [SFP] optical transceiver on the DAG 4.3S card.

The SFP transceiver consists of two parts:

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

The correct transceiver is chosen to suit the optical parameters of the target network installed in the chassis.

The transceiver may then be connected to the network via LC-style optical connectors.

Further information about the Pluggable Optical Transceiver is available at the Endace <http://www.endace.com/dagPluggable.htm> web page.

## 1.3 DAG 4.3S Card Architecture, continued

### Description, continued

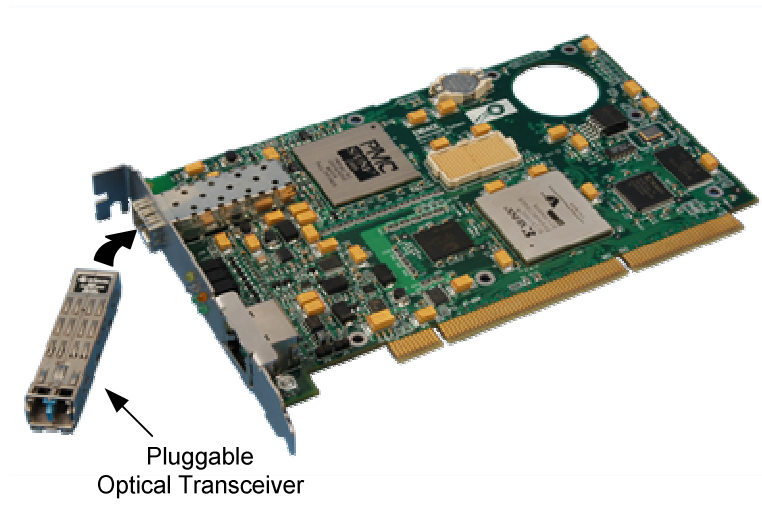


Figure 2-1. Pluggable Optical Transceiver.

## Chapter 3: Setting Optical Power

**Description** The optical power range depends on the particular device fitted on the DAG 4.3S card.

The DAG 4.3S card is shipped fitted with FTRJ 1321S 1300nm single-mode short range optics module by default.

**Optical power measure** Optical power is measured in dBm – decibels relative to 1 mW where 10 dB is equivalent to a factor of 10 in power.

The numbers are all negative, showing powers below 1 mW. The most sensitive devices can work down to about –30 dBm, or 1 uW.

**Configuration** The following table describes the DAG 4.3S card optics power module part, single-mode fibre [SMF], and configuration.

Part No.	Fibre	Data Rate	Min/Max Pwr [dBm] in	Min/Max Pwr [dBm] out	Min/Max Wavelength in	Min/Max Wavelength out
Finisar FTRJ1321S	SMF	2488	-18/0	-9.5/-3	1270/1600	1260/1360
Agilent HFCT5942	SMF	2488	-19/-3	-10/-3	1260/1570	1260/1360

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

- DAG 4.3S Card Optical Power Input
- Splitter Losses

### 3.1 DAG 4.3S Card Optical Power Input

**Description** The optical power input to the DAG 4.3S card must be within a receiver's dynamic range.

When optical power is slightly out of range an increased bit error rate is experienced. If power is well out of range the system cannot lock onto the SONET signal. In extreme cases of being out range excess power will damage a receiver.

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.

### 1.3 DAG 4.3S Card Architecture, continued

**Input power** When the DAG card is set up, measure the optical power at the receiver and ensure that it is within the specified power range.

Input power is adjusted by:

- Changing splitter ratio if power is too high or too low, or
- Inserting an optical attenuator if power is too high.

### 3.2 Splitter Losses

<b>Description</b>	<p>Splitters have the insertion losses marked on packaging or in accompanying documentation.</p> <ul style="list-style-type: none"> <li>• A 50:50 splitter will have an insertion loss of between 3 dBm and 4 dBm on each output</li> <li>• 90:10 splitter will have losses of about 10 dBm in the high loss output, and &lt;2 dBm in the low loss output</li> </ul>
<b>Single mode fibre loss</b>	A single mode fibre connected to a multi-mode input has minimal extra loss.
<b>Multi-mode fibre loss</b>	A multi-mode fibre connected to a single mode input creates large and unpredictable loss.
<b>Wavelength loss</b>	Splitters are designed for a particular wavelength. When mismatched, the split ratio will be different from that which was intended.





## Chapter 4 Confidence Testing

**Introduction** The confidence testing is a process to determine whether the DAG 4.3S card is functioning correctly.

The process also involves a card capture session, and demonstrates configuration in the style of 'What You See You Can Change', WYSYCC.

Interface statistics are also inspected during this process.

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

- Interpreting DAG 4.3S Card LED Status
- DAG 4.3S Card Capture Session
- Configuration in WYSYCC Style
- DAG 4.3S Card Configuration Options
- Inspect Link Data
- Reporting Problems

### 4.1 Interpreting DAG 4.3S Card LED Status

**Description** When a DAG 4.3S card is powered up LED 1 should always illuminate. The other LEDs display for specific functions. The DAG 4.3S has 6 status LEDs, one is coloured blue, one yellow, two green, one orange, and one red.

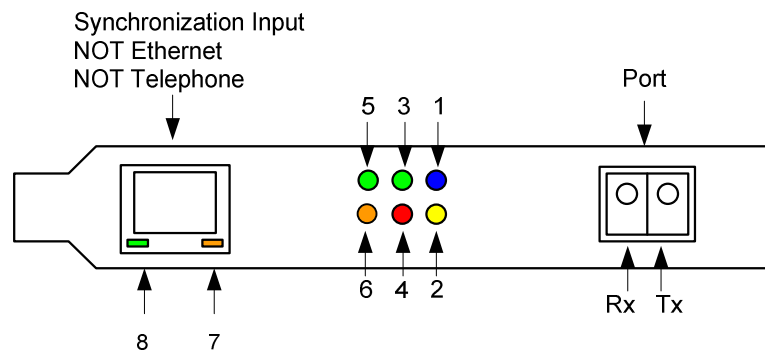


Figure 4-1. DAG 4.3S Card Status LEDs.

## 1.3 DAG 4.3S Card Architecture, continued

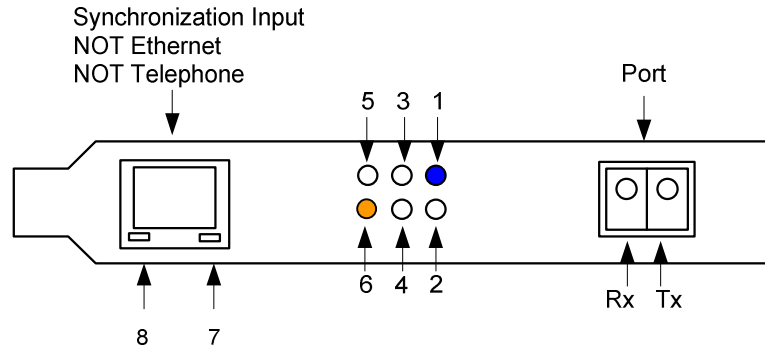


Figure 4-2. LED State for DAG 4.3S Card Without Optical Input.

**LED definitions** The following table describes the LED definitions.

LED	Description
LED 1	FPGA successfully programmed.
LED 2	Data capture in progress.
LED 3	Reserved.
LED 4	Transmitter laser ON.
LED 5	SD. Signal Detect. Illuminates when light is detected.
LED 6	LOF. Loss of frame. Illuminates the framer loses lock on a valid SONET/SDH stream.
LED 7	PPS Out: Pulse Per Second Out – flashes indicate card is sending a clock synchronization signal.
LED 8	PPS In: Pulse Per Second In – flashes indicate card is receiving an external clock synchronization signal.

### Description

The `dagfour` utility supports configuration status and physical layer interface statistics for the DAG 4.3S. In a troubleshooting configuration options `-si` should be passed to the tool to watch the operational status of the optical, SONET and framing layers. More details about the meaning of the various bits are supplied through the help page (`dagfour -h`) as well as via the manual page.

## 4.2 DAG 4.3S Card Capture Session

**Description** The DAG 4.3S uses an ASIC SONET ATM/PoS physical layer interface device to support capturing of ATM cells and HDLC encoded Packet-over-SONET data frames. The card supports both SONET (STS-48c) and SDH (STM-16c).

A successful DAG 4.3S card capture session is accomplished by checking the receiver ports optical signal levels and checking the card has correctly detected the link. This is followed by configuring DAG for normal use.

**Procedure** Follow these steps to troubleshoot DAG 4.3S card configuration.

### Step 1. Check Receiver Ports Optical Signal Levels.

The card supports 1300 nanometer single-mode fibre attachments with optical signal strength between 0 dBm and -18 dBm.

If in doubt, check card receiver ports light levels are correct using an optical power meter.

The card receiver ports are the lower half of the LC-style connector, the closest to the LEDs.

Cover unused card transmit ports with LC-style plugs to prevent dust and mechanical hazards from damaging optics.

### Step 2. Understand Link Layer Configuration

Learn about the link layer configuration in use at the network link being monitored.

Important parameters include specific scrambling options in use. If the information cannot be obtained reliably, the card can be made to work by varying the parameters until data is arriving at the host system.

### Step 3. Check FPGA Image Loaded.

Before configuring the card, ensure the most recent FPGA image is loaded on the card.

```
dag@endace:~$ dagrom -rvp -d dag0 -f xilinx/dag43pcix-terf.bit

dag@endace:~$ dagfour -d dag0
light    nolaser
link     noreset OC48c nofcl noeql
sonet    master scramble
ATM      discard pscramble norxpmts notxpmts
packet   varlen slen=48 noalign64
packetA  drop=0
pcix     133MHz 64-bit buf=128MiB rxstreams=1 txstreams=1 mem=0:0
Firmware: edag43spcix_terf_v2_6 2v1000ff896 2005/10/18 12:45:30 (user)
Card Serial: 3865
```

## 1.3 DAG 4.3S Card Architecture, continued

### Procedure, continued

#### Step 4. Check Card is Locked to Data Stream

Configure card according to local settings.

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

#### Step 5. List Current Settings

For DAG 4.3S framer configuration and statistics the `dagfour` tool is supplied.

Calling `dagfour` without arguments lists current settings.

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

#### Step 6. Configure DAG for Normal Use

The `dagfour default` command is always used:

```
dag@endace:~$ dagfour -d dag0 default
light    nolaser
link     noreset OC48c nofcl noeql
sonet    master scramble
POS      crc32 nocrcstrip short=16 long=1536 discard pscramble rxpkts txpkts
terf     terf_strip32
packet   varlen slen=48 noalign64
packetA  drop=0
pcix     133MHz 64-bit buf=128MiB rxstreams=1 txstreams=1 mem=112:16
Firmware: edag43pcix_terf_v2_6 2v1000ff896 2005/10/18 12:45:30 (user)
Card Serial: 3865
```

#### ATM monitoring

The `default` command always sets the DAG 4.3S card to POS mode. For ATM monitoring use `default atm`.

## 4.3 Configuration in WYSYCC Style

**Description** The configuration of the tool works in WYSYCC style – what you see you can change.

To turn on the card's laser on for instance, type:

```
dag@endace:~$ dagfour -d dag0 laser
light    laser
link     noreset OC48c nofcl noeql
sonet    master scramble
POS      crc32 nocrcstrip short=16 long=1536 discard pscramble rxpkts txpkts
terf     terf_strip32
packet   varlen slen=48 noalign64
packetA  drop=0
pcix     133MHz 64-bit buf=128MiB rxstreams=1 txstreams=1 mem=112:16
Firmware: edag43spcix_terf_v2_6 2v1000ff896 2005/10/18 12:45:30 (user)
Card Serial: 3865
```

**ATM status** In ATM mode, the status output changes:

```
dag@endace:~$ dagfour -d dag0 atm
light    laser
link     noreset OC48c nofcl noeql
sonet    master scramble
ATM      discard pscramble rxpkts txpkts
terf     noterf_strip
packetA  drop=0
pcix     133MHz 64-bit buf=128MiB rxstreams=1 txstreams=1 mem=112:16
Firmware: edag43spcix_terf_v2_6 2v1000ff896 2005/10/18 12:45:30 (user)
Card Serial: 3865
```

For configuration options removing or adding the "no" prefix will change the setting.

## 4.4 DAG 4.3S Card Configuration Options

**Description** There are many DAG 4.3S card configuration options supported.

default	set framer to normal defaults
[no]laser	dis/enable transmit laser
atm	set framer into ATM mode
pos	set framer into Packet-over-SONET (PoS) mode
[no]reset	hold/release framer (in) reset
oc48c	set framer to OC48c mode
[no]fcl	(un)set facility loop back in the phy. This is useful for card chaining
[no]eql	(un)set equipment loop back in the phy
[no]scramble	(un)set SONET scrambling

## 1.3 DAG 4.3S Card Architecture, continued

### Description, continued

master	set card to SONET clock master
slave	set card to SONET clock slave
crc16	PoS CRC16 link
crc32	PoS CRC32 link
[no]crcstrip	Don't include CRC in ERF record or wlen count (DAG 4.2 compatible behaviour).
slen=X	capture X bytes of packet data.
[no]varlen	dis/enable variable length capture. Otherwise record length padded to slen. Defaults to varlen.
short=X	Set minimum expected packet size to X
long=X	Set maximum expected packet size to X
mem=X:Y	Configure memory allocated to streams 0, 1, ...
[no]align64	Generate ERF records with 64-bit alignment [default 32-bit]

### Inspect interface statistics

Once the card has been configured as expected, the interface statistics should be inspected to see if the card is locked to the data stream.

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

### Status bits display

The tool will display a number of status bits as they have occurred since the last time read. In our example, the interval is set to one second via the `-i` option.

LoS	Loss of signal. If set, this indicates that there is either no signal at the receiver or the optical signal strength is too low to be recognized.
OoF	Out of frame. If set, the section overhead processor is not locked to the SONET stream.
LoF	Loss of frame. If set, Oof had been asserted for more than 3 milliseconds.
LAIS	Line alarm indication signal. If set indicates a SONET/SDH remote APS error.
LRDI	Line remote defect indication. If set indicates a SONET/SDH remote APS error.

## 1.3 DAG 4.3S Card Architecture, continued

### Status bits display, continued

LOC	Loss of clock. If set the clock recovery PLL is unable to lock onto and recover the input data stream, check connections and optical signal level.
LOR	Loss of reference. If set the clock recovery PLL is unable to lock onto the reference clock, check configuration.
LoP	Path loss of pointer. If set indicates the path overhead processor is not locked onto the SONET/SDH stream, check connections and optical signal level.
PathBIP	SONET/SDH Path Bit Interleaved Parity error. The link is impaired, check connections and optical signal level.
PathREI	SONET/SDH Path Remote Error Indication. The link is impaired, check connections and optical signal level.

### POS mode

In POS mode, the following columns are present:

RxFrames	Number of PoS frames received since last reading.
RxBytes	Number of PoS payload bytes received since last reading.

### ATM mode

In ATM mode, the following columns are present:

LCD	Loss of Cell Delineation. The framer cannot lock onto the ATM cells.
RxCeLLs	Number of non-idle ATM cells received since last reading.
RxIdles	Number of idle ATM cells received since last reading.

## 1.3 DAG 4.3S Card Architecture, continued

**Extra counters** These extra counters are available with the extended statistics option:

```
dagfour -ei
```

Path\_Label SONET/SDH C2 byte value or Path Signal Label. Typically 0x13 for ATM, 0x16 for PPP, and 0xCF for Cisco HDLC POS.

### When in POS mode

FCS\_Fail Number of PoS frames with FCS errors since last reading.

Aborts Number of PoS frames aborted since last reading.

Short Number of short PoS frames since last reading.

Long Number of long PoS frames since last reading.

RXFDrop Number of PoS frames dropped in the Phy Receive FIFO since last reading.

### When in ATM mode

HEC\_Fail Number of ATM Header Error Checksum Failures since last reading.

### PoS OC-48 stream example

An example for a card locked to a PoS OC-48c stream carrying a constant traffic load is:

LoS	OoF	LoF	LAIS	LRDI	LoC	LoR	LoP	PathBIP	PathREI	RxFrames	RxBytes
0	0	0	0	0	0	0	0	0	0	263097690	33676504219
0	0	0	0	0	0	0	0	0	0	1852309	237095640
0	0	0	0	0	0	0	0	0	0	1865467	238779776
0	0	0	0	0	0	0	0	0	0	1865461	238778954

### Values

The first second has high values as the counters have accumulated their values over more than one second.

### Extended statistics

The same situation shown with extended statistics, with leftmost columns removed for clarity:

RxFrames	RxBytes	FCS Fail	Aborts	Short	Long	RXFDrop	PathLabel
31866294	4078892023	0	0	0	0	0	0xcf
1861661	238292569	0	0	0	0	0	0xcf
1865472	238780385	0	0	0	0	0	0xcf
1865754	238816640	0	0	0	0	0	0xcf



### 1.3 DAG 4.3S Card Architecture, continued

**ATM OC-48C Stream** An example for a card locked to an ATM OC-48c stream carrying a constant traffic load is:

LoS	OoF	LoF	LAIS	LRDI	LoC	LoR	LoP	PathBIP	PathREI	RxCells	RxIdles
0	0	0	0	0	0	0	0	0	0	34298407	33676504219
0	0	0	0	0	0	0	0	0	0	152115	5541287
0	0	0	0	0	0	0	0	0	0	152489	5554947
0	0	0	0	0	0	0	0	0	0	152489	5554931

**Extended statistics** The following OC-48C output is with extended statistics, leftmost columns are removed for clarity.

LoC	LoR	LoP	PathBIP	PathREI	LCD	RxCells	RxIdles	HEC_Fail	PathLabel
0	0	0	0	0	0	6013310	219055106	0	0x13
0	0	0	0	0	0	151186	5507454	0	0x13
0	0	0	0	0	0	152489	5554933	0	0x13
0	0	0	0	0	0	152489	5554941	0	0x13

**POS configuration** The following output shows a problem with optical light levels when in a POS configuration:

LoS	OoF	LoF	LAIS	LRDI	LoC	LoR	LoP	PathBIP	PathREI	RxFrames	RxBytes
1	1	1	0	0	1	0	1	65535	496	3922879	502432705
1	1	1	0	0	1	0	1	16014	0	0	0
1	1	1	0	0	1	0	1	16160	0	0	0
1	1	1	0	0	1	0	1	16158	0	0	0
1	1	1	0	0	1	0	1	16158	0	0	0

**Extended statistics** The following output when in POS configuration is with extended statistics, leftmost columns are removed for clarity.

RxFrames	RxBytes	FCS_Fail	Aborts	Short	Long	RXFDrop	PathLabel
0	0	0	0	0	0	0	0x30
0	0	0	0	0	0	0	0x30
0	0	0	0	0	0	0	0x30
0	0	0	0	0	0	0	0x30

## 1.3 DAG 4.3S Card Architecture, continued

### Stabilise configuration

Follow these steps to stabilise the configuration.

#### Step 1. Ensure Columns are at Zero

Check that the LoS, OoF, and LoF, being the first three columns, are zero.

Check light levels.

#### Step 2. Inspect for BIP Errors

Check that no BIP errors occur, otherwise check cabling and light levels.

#### Step 3 Check CRC Settings

For PoS, ensure scrambling and CRC settings are correct.

#### Step 4. Check REI Errors

REI errors indicate that the remote end of the link is detecting errors.

NOTE: This may not affect the capture of data by the DAG card.

## 4.5 Inspect Link Data

### Description

On Packet-over-SONET (PoS) links it can happen that there is very little or no data information received. This typically indicates incorrect scrambling settings.

While a default is provided that matches typical link settings, the actual configuration varies from network to network.

A remedial action is to vary the scramble and pscramble options and then retry.

If it is necessary to connect the transmit port of the DAG 4.3S to other equipment, it is necessary to enable the transmit laser. The laser normally used is eye safe, but is disabled as a precaution as it is not normally needed.

The laser radiation is in the invisible infrared part of the spectrum. When the laser is turned on, the bright red laser warning LED will be lit.

In a test-bench situation where two DAG cards are connected directly to each other, one card must be designated the SONET clock master, and the other card should be the SONET clock slave, deriving its clock signals from the received network stream.

## 4.6 Reporting Problems

**Description** If there are unresolved problems with a DAG card or supplied software, contact Endace Technical Support via the email address [support@endace.com](mailto:support@endace.com).

Supplying sufficient information in an email enables effective response.

The exact information available to users for trouble, cause and correction analysis may be limited by nature of the problem.

The following problem resolution checklist is applicable to Windows OS system except where indicated for UNIX OS system.

### Problem resolution checklist

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.	UNIX operating system only. Any compiler errors or warnings when building DAG driver or tools.
6.	UNIX operating system only. For Linux and FreeBSD, messages generated when DAG device driver is loaded. These can be collected from command <code>dmesg</code> , or from log file <code>/var/log/syslog</code> .
7.	UNIX operating system only. Output of <code>cat /proc/dag</code> .
8.	Firmware versions from <code>dagbug -cx</code> and <code>dagrom -x</code> .
9.	Physical layer status reported by: <code>dagfour</code>
10.	Network link statistics reported by: <code>dagfour -ei</code>
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 <code>typescript</code> Unix utility may be useful for recording this information.
14.,	A small section of captured packet trace illustrating the problem.





# Chapter 5: Running Data Capture Software

**Introduction** For a typical measurement session, the `scripts/dag43start` script is edited and used to operate the cards directly.

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

- Starting Capture Session
- High Load Performance
- DAG Card Packet Transmission Capabilities

## 5.1 Starting Capture Session

**Description** The various tools used for data capture are in the `tools` sub-directory.

For a typical measurement session, ensure the driver is loaded, the firmware has been downloaded, and the card is configured.

The integrity of the card's physical layer is then set and checked.

**Procedure** Follow these steps to set a data capture session.

Process	Description
Load Xilinx receive image.	<p>For a typical measurement session, first move to the <code>dag</code> directory.</p> <p>Load the driver.</p> <p>Load the Xilinx receive image to each DAG. For example, with two DAG 4.3S cards installed:</p>
	<pre>dagload dagrom -rvp -d dag0 -f xilinx/dag43pcix-terf.bit dagrom -rvp -d dag1 -f xilinx/dag43pcix-terf.bit</pre>
Check integrity.	<p>Set, and then check the integrity of the physical layer to both DAGs as outlined in <a href="#">Chapter 4</a>.</p> <pre>dagfour -d dag0 default dagfour -d dag1 default</pre>

## 1.3 DAG 4.3S Card Architecture, continued

### Procedure, continued

Process	Description
Setting capture session parameters.	<p>Parameters are set with <code>dagfour</code>.</p> <p>The card can operate in two modes, variable length capture (<code>varlen</code>), and fixed length capture (<code>novarlen</code>).</p> <p>In variable length capture mode, a maximum capture size is set with <code>slen=N</code> bytes. This figure should be in the range 32 to 2048 and is rounded down to the nearest multiple of 4.</p> <p>Packets longer than <code>slen</code> are truncated. Packets shorter than <code>slen</code> will produce shorter records, saving bandwidth and storage space. Full packet capture for example:</p> <pre>dagfour -d dag0 varlen slen=1536</pre>
Setting fixed length mode.	<p>In fixed length mode, packets longer than the selected <code>slen</code> are truncated to <code>slen</code>.</p> <p>Packets shorter than <code>slen</code> produce records padded out to <code>slen</code> length.</p> <p>Large <code>slen</code> values in fixed length mode should not be used because short packets arriving produce large padded records, wasting bandwidth and storage space.</p> <p>An example, for fixed length 64-byte records, choose <code>slen=48</code> (64 – ERF header size of 16): is:</p> <pre>dagfour -d dag0 novarlen slen=48</pre>

## 1.3 DAG 4.3S Card Architecture, continued

### Procedure, continued

Process	Description
Starting a capture session.	<p>Once the capture parameters are configured, a capture session is started by:</p> <pre>dagsnap -d dag0 -v -o tracefile0 &amp; dagsnap -d dag1 -v -o tracefile1</pre> <p>Option <code>-v</code> provides user information during capture; it can be omitted for automated trace runs.</p> <p>If the <code>-o tracefile</code> parameter is not specified the tool writes to stdout, which can be used to pipeline <code>dagsnap</code> with other tools from <code>dagtools</code> package.</p> <p>By default <code>dagsnap</code> runs forever.</p>
Stopping <code>dagsnap</code>	<p><code>dagsnap</code> can be stopped with a signal:</p> <pre>killall dagsnap</pre> <p>or key strokes <code>CTL + C</code></p> <p><code>dagsnap</code> can also be configured to run for a fixed number of seconds and then exit using the <code>-s</code> option.</p>

## 5.2 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.

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

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

is displayed on the PC screen, and printed to `log /var/log/messages`. The "Data capture" LED also goes out. This may be visibly indicated as flashing or flickering.



## 1.3 DAG 4.3S Card Architecture, continued

**Detecting packet losses**      Until some data is read out of the buffer to free some space, any arriving packets subsequently will be discarded by the DAG card. Any loss can be detected in-band by observing the `lctr` [Loss Counter] field of the Endace Record Format.

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

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. Capture at OC-12/STM-4 (622Mbps) rates and above may require a larger buffer.

128MB or more is suggested for Linux/FreeBSD.

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 must be less than 890MB for Linux on 32-bit platforms.

## 5.3 DAG Card Packet Transmission Capabilities

**Description** The firmware included with the DAG 4.3S card allows the DAG to transmit as well as receive packets, however the DAG does not appear as a network interface to the operating system.

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

- DAG Packet Transmission
- Inline Forwarding

### 5.3.1 DAG Packet Transmission

**Process** The following information describes the DAG capabilities of the DAG firmware for the transmission and receiving of packets.

Process	Description
Explicit packet transmission.	<p>The DAG will not respond to ARP, ping, or router discovery protocols. It will only transmit packets explicitly provided by the user.</p> <p>This capability allows the DAG card to be used as a simple traffic load generator.</p> <p>The DAG can also be used to retransmit previously recorded packet traces.</p> <p>By default, the packet trace will be transmitted at 100% line rate, the packet timing of the original trace file is not reproduced.</p>
Packet transmission utility.	<p>The <code>dagflood</code> utility can transmit ERF format packet traces. The ERF trace file to be transmitted must contain only ERF records of the type matching the current link configuration.</p> <p>The ERF records to be transmitted must all have a length which is a multiple of 64-bits. When capturing a packet trace for later transmission, you can set 64-bit alignment using the <code>dagfour align64</code> command.</p>

Error! No text of specified style in document., **continued**

**Process**,continued

<b>Process</b>	<b>Description</b>
Convert trace files.	<p>It is also possible to convert trace files that have been captured without the <code>align64</code> option. This can be done with the command:</p> <pre>dagconvert -v -i in.erf -o out.erf -A8</pre> <p>If uncertain that a trace file is 64-bit aligned for transmission with <code>dagflood</code>, the file can be tested with <code>dagbits</code>:</p> <pre>dagbits -vvc align64 -f tracefile.erf</pre> <p>If a captured trace file is not available, the <code>daggen</code> program is capable of generating trace files containing simple traffic patterns. This allows the DAG card to be used as a test traffic generator.</p>
Capture received traffic while transmitting.	<p>It is possible to capture received traffic while transmitting.</p> <p>Capture programs such as <code>dagsnap</code>, <code>dagconvert</code>, and <code>dagbits</code> can be used while <code>dagflood</code> is sending packets.</p> <p>Use of 133MHz PCI-X is recommended to ensure adequate bandwidth is available for simultaneous receive and transmit operation.</p>

Error! No text of specified style in document., **continued**

**Process** (continued)

<b>Process</b>	<b>Description</b>
Configuring DAG card for transmission.	<p>To configure a DAG card for transmission, some memory must be allocated to a transmit stream.</p> <p>In the <code>dagfour</code> output, <code>buf=nMB</code> indicates that <code>n</code> megabytes of memory has been allocated to this DAG card in total. This memory can be split between the available receive and transmit stream buffers.</p> <p>The memory allocation is displayed with <code>mem=X:Y</code>, where <code>X</code> is the amount of memory allocated to receive stream 0 in MB, and <code>Y</code> is the amount of memory allocated to transmit stream 1 in MB.</p> <p>By default the memory is evenly split between the receive streams, the transmit streams have no memory allocated.</p> <p>If the card is to be used only for transmit, the <code>dagfour txonly</code> option can be used to recover the receive buffer memory and assign all the memory to transmit.</p> <p>If the card is to be used for both transmitting and receiving, the <code>rxtx</code> option can be used. This allocates 16MB of memory to each transmit stream, and divides the remaining memory between the receive streams. Alternatively the memory allocation can be directly set with <code>mem=X:Y</code> option.</p> <p>The stream buffer memory allocation can only be changed when no packet capture or transmission programs are running.</p>

### 5.3.2 Inline Forwarding

**Description** The DAG card can be used as an 'inline' device to receive, inspect, filter and forward packets between Port A and Port B.

**Process** The following information describes the DAG card inline forwarding process.

<b>Process</b>	<b>Description</b>
Inline transmission.	This operation can be performed at 100% line rate in both directions simultaneously. A PCI-X 133MHz slot is required for full performance and the performance may be limited by the host PC CPU and memory performance.
The <code>dagfwddemo</code> program.	<p>The <code>dagfwddemo</code> program is provided as a demonstration of how this can be achieved. This program forwards packets bidirectionally, applying a user supplied BPF filter to each packet with the host CPU.</p> <p>Packets which match the filter are forwarded, while packets that do not match are dropped.</p>
Modification of packets.	<p>Modification of packets during inspection is also possible.</p> <p>The modifications should not change the length of the packet, and the user is responsible for re-computing checksums as needed.</p> <p>This is intended a demonstration of Inline Forwarding technology for use in firewall or IDS/IPS applications. It is not suitable for use as a production firewall.</p>



## Chapter 6: Synchronizing Clock Time

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

The system that provides 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. A card's clock is initialised using the PC clock, and then free-runs using a crystal oscillator.

Without synchronization, 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 synchronization connector, or the host PCs clock can be used in software.

**Common synchronization** The DAG card 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 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

## 6.1 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 synchronization from several input sources, and can also be made to drive its synchronization output from one of several sources.

Synchronization settings are controlled by the `dagclock` utility.

### Example

```
dag@endace:~$ dagclock -h
Usage: dagclock [-hv] [-d dag] [option]

    -h  this page
    -v  increase verbosity
    -d  DAG device to use

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 synchronization 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
```

## 6.2 Time Synchronization Configurations

**Description** The DUCK is used in several ways, with or without an external time reference source.

Uses include 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



## 6.2.1 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 PCs 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 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
```

**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.

## 6.2.2 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.

Error! No text of specified style in document., **continued**

**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 synchronization 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:

```
dagclock -d dag0 none overout
```

**Preventing time-stamp drift**

To prevent DAG card time-stamps drifting against UTC, one card is synchronized to the host PC clock, which in turn utilises NTP. This provides a master signal to the second card.

In this case, connect synchronization connectors 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    overin
muxout   overout
```

### 6.2.3 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 synchronization 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 [Time Distribution Server] modules which allow multiple DAG cards to use a single receiver.

Error! No text of specified style in document., **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.

Time accuracy is achieved using the host PC NTP. The time reference source is connected to synchronization connector, the card automatically synchronizes to a valid signal.

**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 and existing RJ-45 cabling infrastructure.

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

**Testing signal** For Linux and FreeBSD, when a synchronization 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 dagN
```

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

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

### 6.3 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 DAG 4.3S card 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-

### 5.3 DAG Card Packet Transmission Capabilities, continued

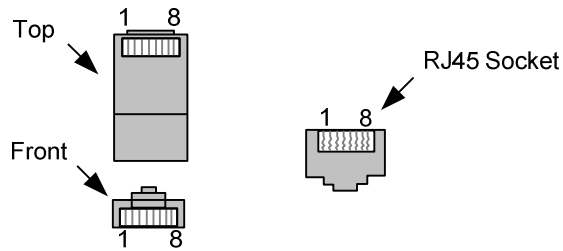


Figure 6-1. RJ45 Plug and Socket Connector Pin-outs.

#### 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 DAGs 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](mailto:support@endace.com).



## Chapter 7: Data Formats

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

- Data Formats
- Timestamps

### 7.1 Data Formats

**Description** The DAG 4.3S uses the ERF Types 1 and 3 timestamps.

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.

timestamp		
timestamp		
type:1	flags	rlen
lctr		wlen
HDLC Header		
(rlen - 20) bytes of packet		

Table 7-1. Type 1 PoS HDLC Variable Length Record.

timestamp		
timestamp		
type:3	flags	rlen
lctr		wlen
ATM Header		
48 bytes of cell		

Table 7-2. Type 3 ATM Cell Record.

## 7.1 Data Formats, continued

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

<b>Data Format</b>	<b>Description</b>
type:	<p>This field contains an enumeration of the frame subtype. If the type is zero, then this is a legacy format.</p> <p>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            8: TYPE_MC_RAW_CHANNEL: Multi-channel Raw Link Data.            9: TYPE_MC_AAL5: Multi-channel AAL5 frame            10: TYPE_COLOR_HDLC_POS: HDLC POS like TYPE_HDLC_POS, but with the LCNTR field reassigned as COLOR            11: TYPE_COLOR_ETH: Ethernet like TYPE_ETH, but with the LCNTR field reassigned as COLOR</p>
flags:	<p>This byte is divided into 2 parts, the interface identifier, and the capture offset.</p> <p>1-0: Enumerates capture interface 0-3            2: Varying record            3: Truncated record [insufficient buffer space]            4: RX Error [link layer error]            5: DS Error [internal error]            7-6: Reserved</p>
Rlen: record length	Total length of the record transferred over PCI bus to storage.

## 7.1 Data Formats, continued

### Data format, continued

Data Format	Description
Lctr: <i>loss counter</i>	<p>A 16 bit counter, recording the number of packets lost since the previous record between the DAG card and memory hole due to overloading on PCI bus.</p> <p>The counter starts at zero, and sticks at 0xffff.</p>
Wlen: <i>wire length</i>	<p>Packet length including some protocol overhead. The exact interpretation of this quantity depends on physical medium.</p>

## 7.2 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 is also available to Windows users 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.



## 7.1 Data Formats, continued

**Example code** Here is some example code showing how a 64-bit ERF timestamp (erfts) can be converted into UNIX 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;
}
```

---