
# IA Title: Serial Look Aside Interface Implementation Agreement

**IA # OIF-SLA-01.0**

***December 11, 2007***

Implementation Agreement created and approved
by the Optical Internetworking Forum
www.oiforum.com

**Working Group: Physical and Link Layer Working Group**

---

**TITLE: Serial Look Aside Interface Implementation Agreement**

---

**SOURCE:**

**David Carr**
IDT Inc.
450 March Road
Ontario, K2K 3K2
Canada
Phone + 1 613 287-5127
Email: david.carr@idt.com

**Harmeet Bhugra**
IDT Inc.
6024 Silver Creek Valley Road
San Jose, CA, 95138
USA
Phone + 1 408 284-2687
Email: Harmeet.Bhugra@idt.com

**DATE: December 11, 2007**

---

**ABSTRACT:** This specification defines the Serial Look-Aside Interface implementation agreement (SLA). The interface supports the transfer of command, data, result, and maintenance traffic between a host controller (ASIC, FPGA, or NPE) and a look-aside co-processor. The interface is not targeted to a particular bandwidth it is intended to support a range of co-processors and data rates. The interface also supports the transfer of status and maintenance information in band with the commands. This can either be Credit Pool Status information or Signaling Data Transfers, utilizing NPF messaging.

---

OIF-SLA-01.0

# Serial Look Aside Interface

**The OIF is an international non profit organization with over 130 member companies, including the world's leading carriers and vendors. Being an industry group uniting representatives of the data and optical worlds, OIF's purpose is to accelerate the deployment of interoperable, cost-effective and robust optical internetworks and their associated technologies. Optical internetworks are data networks composed of routers and data switches interconnected by optical networking elements.**

**With the goal of promoting worldwide compatibility of optical internetworking products, the OIF actively supports and extends the work of national and international standards bodies. Formal liaisons have been established with The ATM Forum, IEEE 802.3, IETF, ITU-T Study Group 13, ITU-T Study Group 15, MEF, NPF, ATIS-TMOC, ATIS-OPTXS, TMF, UXPi and the XFP MSA Group.**

For additional information contact:
The Optical Internetworking Forum, 48377 Fremont Blvd.,
Suite 117, Fremont, CA 94538
510-492-4040 Φ info@oiforum.com

www.oiforum.com

# 1 Table of Contents

## 2 List of Figures



## 3 List of Tables

# 4 Document Revision History

| Revision | Date | Revision |
|---|---|---|
| 3 | 16/08/2007 | Straw Ballot candidate |
| 4 | 10/17/2007 | Incorporate Straw Ballot comments |

# 5 Introduction

## 5.1 Overview

This document specifies the Serial Look-Aside Interface (SLA), an interface for the connection of network processing elements (FPGAs, ASICs, and Network Processors) and look-aside coprocessors (e.g., classification devices or memory).

Typical applications may include the transfer of portions of packet data from a NPE for processing by a companion device (offload) and the return of resultant data. The interface is transactional in nature (in contrast to the streaming nature of SPI-S). For example, this interface includes support for Network Search Engines (NSEs) and Smart Memory devices (e.g. Statistics Engine).

The SLA specification borrows heavily from the SPI-S specification [1] for link-level requirements, including data framing and packet delineation, flow control, address formats, and error detection. SLA is derived from, but not strictly compatible with SPI-S. ***This document is intended to capture only the relevant differences between SPI-S and SLA.***

Traditional Look-Aside applications are evolving to become Serial as:

- Number of pins on ASSPs and NPUs is growing too fast
- Routing of parallel interfaces is becoming very complex in support of increasing line rates
- Lack of bandwidth on standardized co-processor interfaces
- Dedicated NPE pins restrict the mix and match of co-processors to applications This is body text for all sections. This is body text for all sections. This is body text for all sections. This is body text for all sections. This is body text for all sections.

SLA reduces the number of pins and enable higher bandwidth interface for Look Aside applications. At the same time, SLA enables NPE interface pins to be allocated to different co-processors depending on the application. Both the number of co-processors and the bandwidth to each co-processor can be tailored as required by the platform.

The protocol operates over 64/66B encoded serial links. The interface is not tied to a particular physical interface or any fixed bandwidth and is capable of operating over individual serial links or multi-lane aggregated links.

The SLA specification defines the link-level requirements, including data framing and packet delineation, flow control, address formats, and error detection. Configuration of SLA may be implemented either in-band or out of band. The interface is defined to operate in a self-healing manner in the face of errors through the use of soft state, but does not include a retry recovery mechanism for payload protection.

### 5.2 Requirements

Serial Look-Aside Interface:

- Shall be point-to-point
- Shall include support for nx4 Serial lanes at 10-100 Gbps aggregate data rate.
- Shall support 8" (20 cm) of FR4 and 1 connector.
- One connector per SLA subsystem
- SLA only defines the interface between the host (NPE) and the slave (co-processor)
    - SLA does not define slave-to-slave I/F
    - SLA does not define host-to-host I/F
- Target for SLA latency:
    - Single slave: less than 200 ns

The SLA protocol shall allow for future upgrade of bandwidth performance by increasing the SERDES speed. The serial protocol includes a CRC function of sufficient capability to support error detection of all single bits in the full protocol. The SLA interface will be a Request/Response interface:

- Common Request/Address bus
- Response bus (i.e. slave pushes results to host - no need to issue another read to retrieve results)

### 5.3 Scope and Purpose

This specification does NOT attempt to encompass applications that could be handled efficiently using SPI-S. For example, co-processors that handle interleaved streams of traffic from many sources or context may be better served

by SPI-S than through SLA. The rationale is to simplify the co-processor implementation wherever possible.

## 5.4 Typical Applications

### 5.4.1 System Architecture Examples

Figure 5-1 is a reference diagram illustrating a typical application utilizing the SLA.

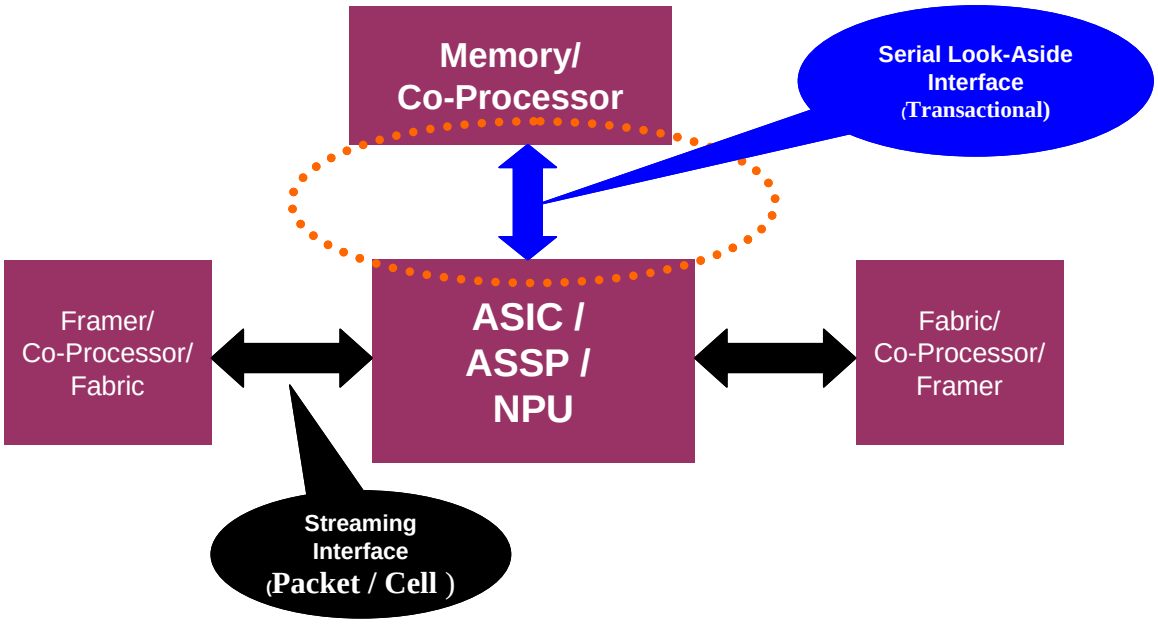


**Figure 5-1 System Level Architecture Example**

### 5.4.2 SLA Configurations

SLA supports both cascade and ring architectures (see Figure 5-2). Because the SLA specifies only the interface between the host and the first adjacent co-processor device, interconnect strategies for slave-to-slave connections in either the cascade or ring topologies is application dependent. Alternative topologies are possible and not prohibited by SLA.

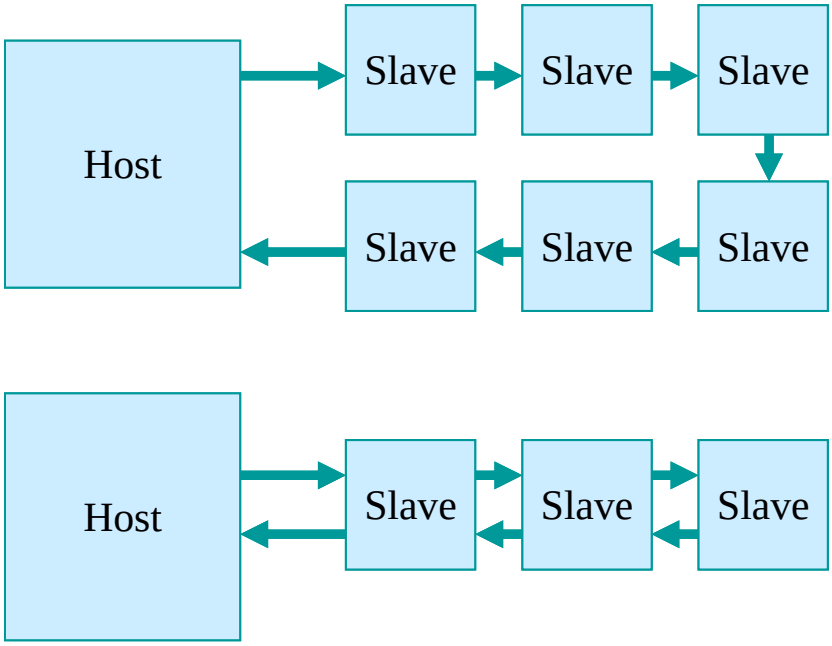


**Figure 5-2 Ring and Cascade Architectures**

## 5.5 References

The following documents contain provisions, which through reference in this text constitute provisions of this specification. At the time of publication, the editions indicated were valid.

All referenced documents are subject to revision. Parties implementing this SLA specification MUST conform to the revisions listed below even if a newer specification become available.

[1] OIF-SPI-S-01.0, Scalable System Packet Interface Implementation Agreement: System Packet Interface Capable of Operating as an Adaptation Layer for Serial Data Links

[2] IEEE-802.3ae-2002

Portions of the OIF SPI-S [1] specification were used to create this document.

## 5.6 Specification Conventions

This specification follows the following protocol of terminology:

SHALL indicates that the item is a requirement for conformance to this specification.

MAY indicates that the item is optional.

SHOULD indicates that the item is not required by this specification, but is offered as implementation guidance.

In addition:

If there is a conflict between the SPI-S specification and the text of this document then the SPI-S specification takes precedence over this document.

Figures (except state diagrams) are informative only.

Footnotes used in this specification are informative only.

A reserved field or bit shall be transmitted as zero and shall be ignored on reception.

### 5.7 Definitions

This specification makes use of defined terms in the SPI-S document. However, the terms below are either modified from the SPI-S document or are new:

Network Processing Element (NPE):

> Any device that uses the SLA as a look-side interface to a co-processor. This differs from the SPI-S definition that groups co-processors into the NPE class and uses SPI-S for communication.

Payload Data:

> Data on the SLA link that is part of a transaction Request or Response.

Response:

> Any solicited data or indication returned by the Slave (co-processor) as a result of transaction Request made by the Host (NPE).

Request:

> A command issued to the Slave (co-processor) by the Host (NPE).

# 6 Architectural Overview

The SLA provides for the transfer of data traffic and flow control information between NPE and co-processor device. The framing format supported for the data framing is described in the following section.

## 6.1 SLA Reference Model

The SLA supports one framing format defined below and illustrated in Figure 6-1:

- 64B/66B: 10 Gigabit Ethernet compatible framing and scrambling. A mandatory mode of operation utilizing 64B/66B coding and scrambling, as defined in IEEE802.3ae-2002 [2]:
  - Clause 49.1.4.1 - PCS introduction
  - Clause 49.2 - Physical Coding Sub layer

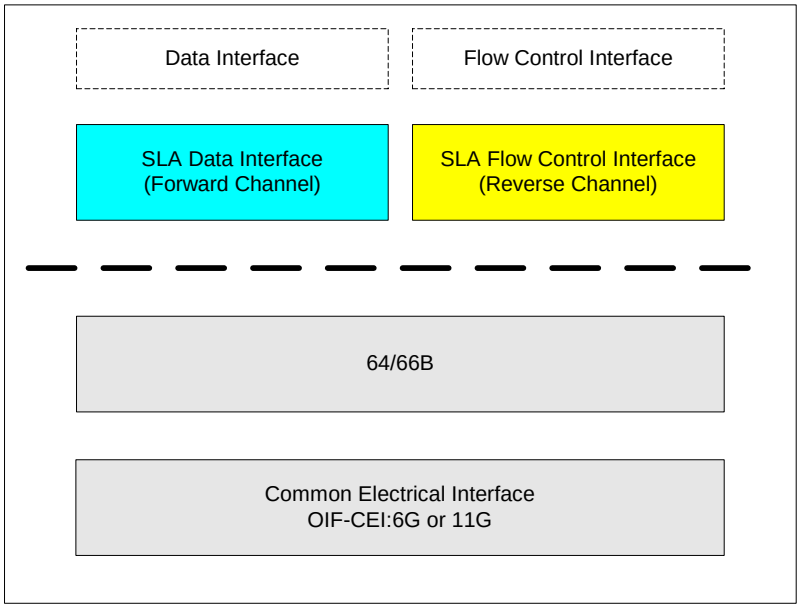


**Figure 6-1 Reference Model**

### 6.1.1 Features of SLA

The SLA has the following characteristics:

- It can efficiently transfer a range of transaction sizes, such as:
  - Memory reads or writes
- Address resolution requests/responses (i.e. MAC/IP lookup)
- Access Control List/QoS Classification requests/responses (N-tuple)

The logical operation is based on the SPI-S specification.

SLA Links may be uni-directional, bi-directional or asymmetric.

## 6.2 Interface Description

An SLA interface is not limited to a particular physical interface. The physical format can use any number of links operating at any application specific frequency. Information transfer a may be uni-directional as well as bi-directional. SLA links may be symmetrical and bi-directional (Figure 6-2), uni-directional (Figure 6-3) or asymmetric (Figure 6-4), for example. In a full duplex implementation both the Host (NPE) and Slave (co-processor) interfaces carry a forward channel (Payload Data) and optionally reverse channel (Signaling) information associated with requests and responses as illustrated in Figure 6-2. Payload Data Transfers are delimited by Start and End of Packet Control Words. These packets may be further divided into segments and flow control information for the reverse channel may be optionally interleaved with the Payload Data.

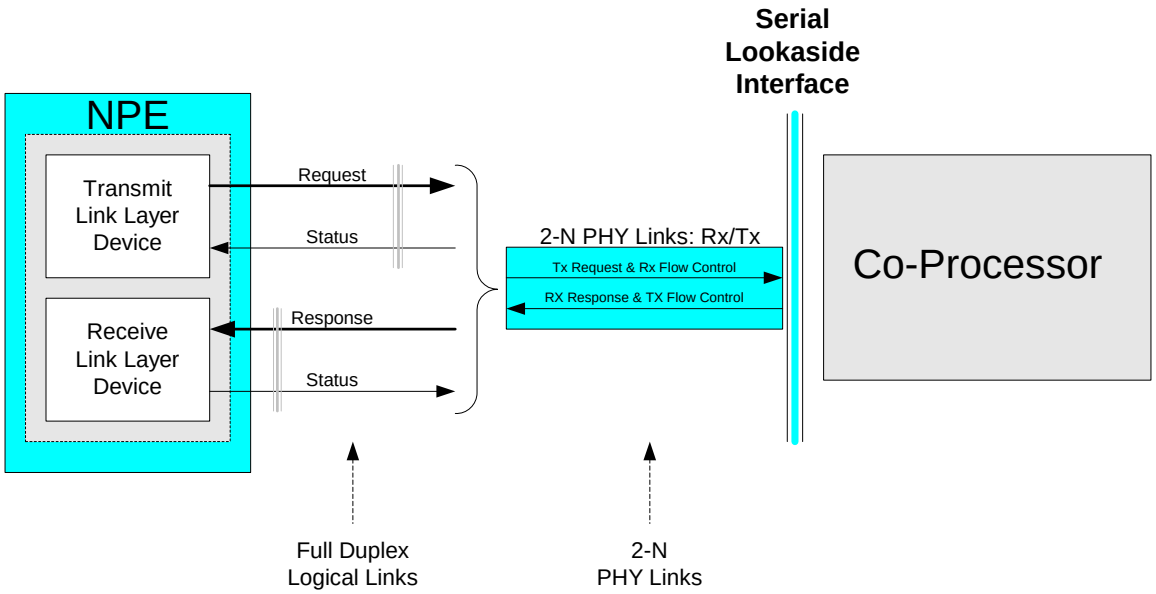


**Figure 6-2 SLA Bi-directional**

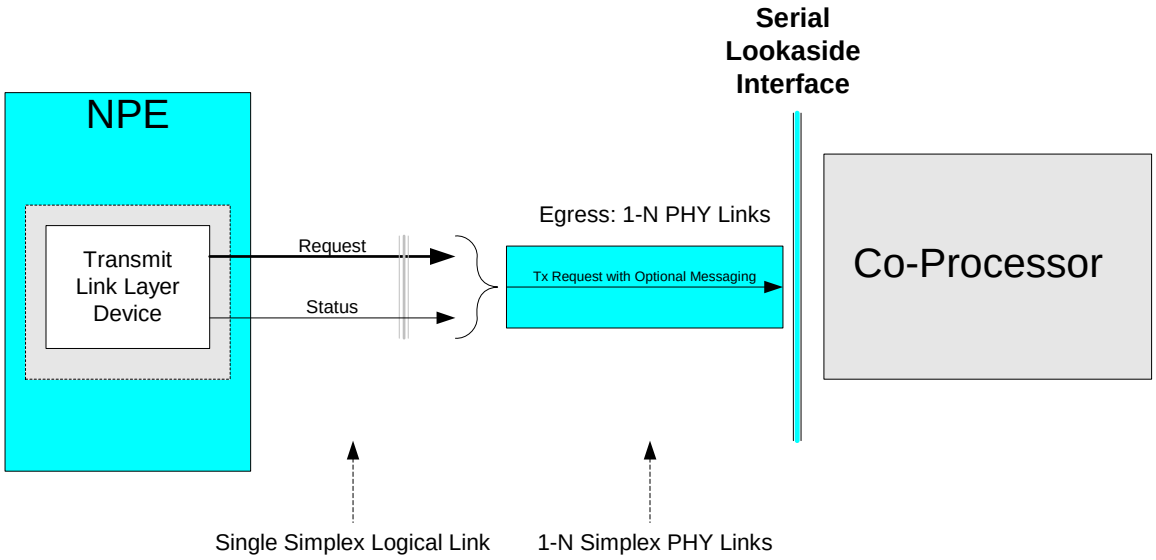


**Figure 6-3 SLA Uni-directional**

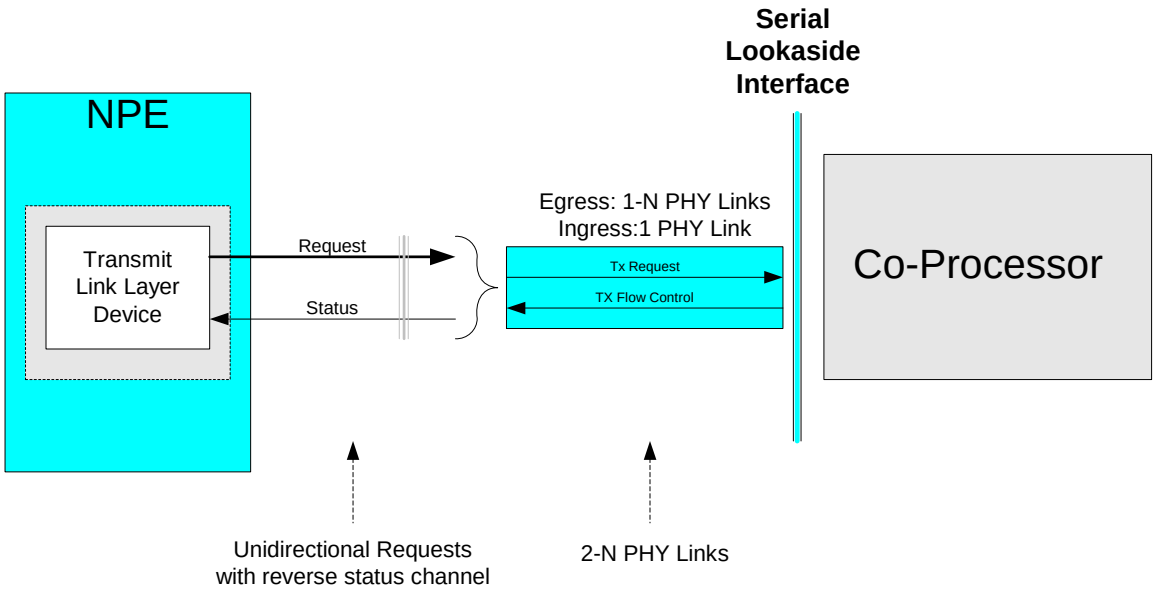


**Figure 6-4 SLA Asymmetrical**

# 7 Data Path Operations

The data path operation of SLA is identical to SPI-S with the following exceptions:

The segment length is modified to support finer granularity of requests, allowing for higher efficiency for small transfers.

Only SPI-S Packet Interleave Mode is supported. Segment Interleave Mode is NOT supported.

The SPI-S Suspend feature is NOT supported.

Because a limited number of channels are supported over the SLA interface, portions of the control word address field are redefined to convey a context identifier associated with the request or response.

A common reference clock MUST be supplied to both the NPE and co-processor for proper SLA operation.

These differences are explained in the following sections.

### 7.1 Segment Length

As with SPI-S (see section 6 of [1]), segmentation of Payload Data Transfers shall only occur at segment boundaries. A segment is of length L (or a multiple of L) bytes. L is a link provision-able parameter, which is an integer multiple (m) of 16 bytes.

**For SLA, "m" MUST BE in the range 1 <= m <= 8.**

NPEs MAY chose to support the entire range of "m" (i.e. transfers from 128 to 1024 bits) to enable flexibility in connection to a variety of co-processors. Co-processors need only implement the specific range required for their operation.

As with SPI-S, a PDT may be any integer multiple of bytes. This is accommodated by termination of the packet with End of Packet Padded (EPAD).

### 7.2 Interleaving of Segments

**The SLA supports only SPI-S Packet Interleave Mode** (see section 6.1 of [1])**.** In Packet Interleave Mode only one active or paused Payload Data Transfer is allowed. Refer to the SPI-S document for Packet Interleave Mode operation.

### 7.3 Suspend

**Due to the limited range of segment lengths supported, SLA does NOT support the SPI-S Suspend feature** (see section 6.1 of [1])**.** All SPI-S control word values and state transitions associated with the Suspend feature are invalid in SLA.

### 7.4 Control Block Format

The Control Word is similar to SPI-S (see section 6.3 of [1]), containing the same five control flags [4:0] and a CRC field [11:0]. These are explained in detail in the SPI-S document.

However, the address field for Payload Data Transfers (PAYLOAD =1) has been redefined into 2 new fields:

- A 4-bit logical channel identifier CHAN[3:0];
- An 11-bit context identifier CONTEXT[10:0];

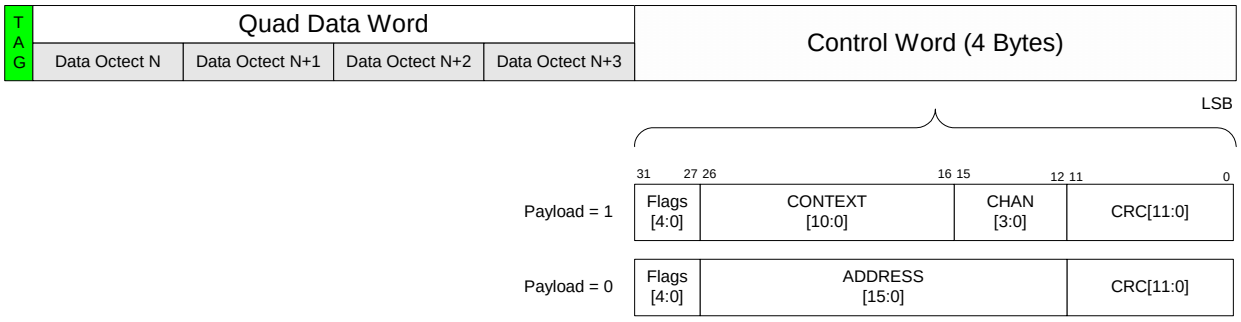


**Figure 7-1 Control Block Format**

#### 7.4.1 ADDRESS Field of Control Word

The function of the address field is determined by the state of the Flag bits in the same control word. If Payload Flag is active then the associated data is a Payload Data Transfer, and the address field consists of 2 parts:

- A 4-bit logical channel address
- A 11-bit context ID

The number of Credit pools is thus restricted to 16 in SLA. Un-provisioned Credit Pools should not be addressed.

If Payload Flag is not active then the associated data is either Signaling Data, Pool Status, Sync or Idle as defined in SPI-S and the address field meaning remains unchanged from SPI-S (see 6.3.5 of [1]).

For Status Data the Control Word Address Field is used to differentiate between the various types of Status Data and is summarized in Table 1.

| DATA TYPE | CONTROL WORD FLAG(s) | ADDRESS | |
|---|---|---|---|
| PAYLOAD | PAYLOAD | ADDRESS[3:0] = Logical Channel | ADDRESS[14:4] = Context ID |
| SIGNALING | NOT PAYLOAD | ADDRESS [3:0] = "0100" | ADDRESS[14:4] = reserved |
| IDLE | NOT PAYLOAD | ADDRESS[14:0] ="000_0000_0000_0000" | |
| POOL STATUS | NOT PAYLOAD | ADDRESS [14:0] = "000_0000_0000_0010 | |
| SYNC | FLAG[4:0] = 10101 & After = 1 | ADDRESS[0] = 1<br>ADDRESS[7:1] = Number of Bit Lanes in Link<br>ADDRESS[14:8] = Current Bit Lane (starting with 1) | |

**Table 1 Status Data Address Decoding**

Note that since there are at most 16 Credit Pools the address for Pool Status is fixed because the status for all pools is conveyed in a single quad-word.

### 7.4.2 Context ID

The context ID is an opaque field used to correlate requests from the host to responses returned by the co-processor. For every request requiring a response, the co-processor must return this field exactly as sent by the host.

### 7.4.3 Logical Channels

To support chains or rings of co-processor attached to one SLA host port, up to 16 logical channels may be defined. These logical channels may for example be used to address physical devices, or to address logical units within a physical device. SLA does not mandate the use of these channels.

### 7.4.4 PDT Contents

The entire Payload contents are application dependent. Each co-processor can chose to implement unique instruction sets.

### 7.5 Clocking

SLA requires the use of a synchronous clocking scheme. Both the NPE and co-processors MUST be supplied with a common reference clock as shown in Figure 7-2.

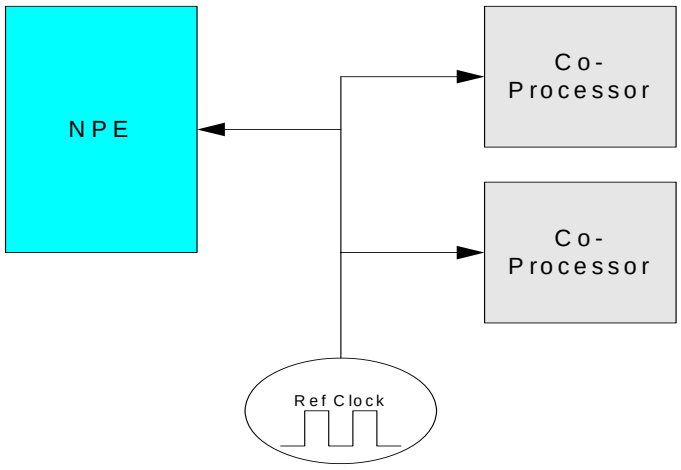


**Figure 7-2 Common Reference Clock**

# 8 Appendix A: List of Companies belonging to the OIF when Document was Approved

ADVA Optical Networking
Agilent Technologies
Alcatel-Lucent
Altera
AMCC
Analog Devices
Anritsu
AT&T
Avago Technologies Inc.
Avanex Corporation
Bookham
Booz-Allen & Hamilton
Broadcom
BT
China Telecom
Ciena Corporation
Cisco Systems
ClariPhy Communications
CoreOptics
Cortina Systems
Data Connection
Department of Defense
Deutsche Telekom
Ericsson
Finisar Corporation
Flextronics
Force 10 Networks
Foxconn
France Telecom
Freescale Semiconductor
Fujitsu
Furukawa Electric Japan
Huawei Technologies
IBM Corporation
IDT
Infinera
Intel
IP Infusion
JDSU
KDDI R&D Laboratories
Level 3 Communications
LSI Logic
Marben Products
MergeOptics GmbH
Mintera
MITRE Corporation
Mitsubishi Electric Corporation
Molex
NEC
NeoPhotonics
Nokia Siemens Networks
Nortel Networks
NTT Corporation
Opnext
PMC Sierra
Sandia National Laboratories
Santur
Sierra Monolithics
Silicon Logic Engineering
Sycamore Networks
Syntune
Tektronix
Telcordia Technologies
Telecom Italia Lab
Tellabs
Texas Instruments
Time Warner Cable
Transwitch Corporation
Tyco Electronics
Verizon
Vitesse Semiconductor
Yokogawa Electric Corporation
ZTE Corporation