PACKET OVER SONET

Implementing Packet Based SONET/SDH Links

WHITE PAPER

TABLE OF CONTENTS

LIST OF FIGURES

1. Introduction

Packet-Over-SONET/SDH (POS) is an emerging technology for carrying IP and other data traffic over the SONET/SDH[1] backbone. Variable length data packets are mapped directly into the SONET Synchronous Payload Envelope (SPE). It may be used in layer 2 switches or layer 3 switches/routers depending on the specific implementation. POS provides reliable, high capacity, point-to-point data links using the SONET physical layer transmission standards.

Mapping into SONET using the Point-to-Point Protocol (PPP) was standardized in accordance with RFC 1619, *"PPP over SONET/SDH,"* and RFC 1662, *"PPP in HDLC-like Framing"*, at OC-3/STM-1, OC-12/STM-4 and OC-48/STM-16 rates. Other packet mappings into SONET such as Frame Relay or Ethernet are not yet standardized but will be discussed as they are currently being implemented in proprietary applications.

2. Network Overview

The explosive demand for bandwidth is motivating Internet Service Providers (ISPs) and corporate networks to upgrade their Internet backbone networks with SONET transmission facilities. These high-speed links not only increase capacity, but also provide operations, administration and management functions that increase network reliability.

2.1 SONET Transport

SONET is a world-wide standardized transmission protocol for implementing a robust, scalable transport mechanism with industry standardized interfaces. It provides a standard operating environment with defined protocols for operations management, provisioning, and performance assurance.

SONET equipment is being deployed into the field by a large variety of service providers. SONET equipment is being used to transport DS-1, E1, DS-2, E2 (through virtual tributaries), DS-3, E3, E4, ATM, and other services such as PPP encapsulated IP datagrams. Both point-to-point and ring-based architectures are being implemented. Service providers have installed SONET equipment in most new carrier systems and transport facilities. This trend will continue for the foreseeable future.

SONET transmission facilities, then, prove to be ideally suited to not only transport ATM cells, but other packet types like IP through the emergence of POS technology. POS is applicable to both layer 2 and layer 3 switching. A layer 3 implementation maps IP datagrams from a multi-protocol router into SONET. These use PPP, HDLC and finally SONET as the layer 2 and layer 1 protocols. Layer 3 routing is only performed and no

[1] Hereby referred to as SONET

layer 2 switching is provided. The SONET network can provide the layer 1 connectivity that is provisioned when the connection is initially setup (i.e. leased line connection). POS point-to-point data links can be implemented over the current SONET standardized line rates. Figure 1 shows the common rates and where POS is likely to be used.

OC Level	STS Level	Line Rate (Mbps)	Packet over SONET Usage
OC-1	STS-1	51.840	Unlikely for direct mapping; however, packets could be mapped into DS-3s then into STS-1s and multiplexed into OC-3's, OC-12's etc..
OC-3	STS-3c	155.520	Access for ISP/Corporate/Frame Relay
OC-12	STS-12c	622.080	Backbone for ISP/Corporate/Frame Relay
OC-48		2488.320	Carrier ISP, OC-48 backbone
OC-192		9953.280	Future

Figure 1. SONET/SDH Signal Hierarchy Common Rates

The POS stream must be inserted into the SONET/SDH Synchronous Payload Envelope on a continuous basis. As packed based services are asynchronous by nature, a method is needed to transform the non-continuous packet stream into a continuous stream. This rate coupling is accomplished by inserting a flag sequence during idle periods in the packet stream. Because POS frames have a variable length, they do not have a specific alignment within the SPE. A POS Frame, therefore, can spread over two SONET frames. As the STS-12c SONET rate is a likely candidate for volume POS deployment, this mapping is illustrated in Figure 2.

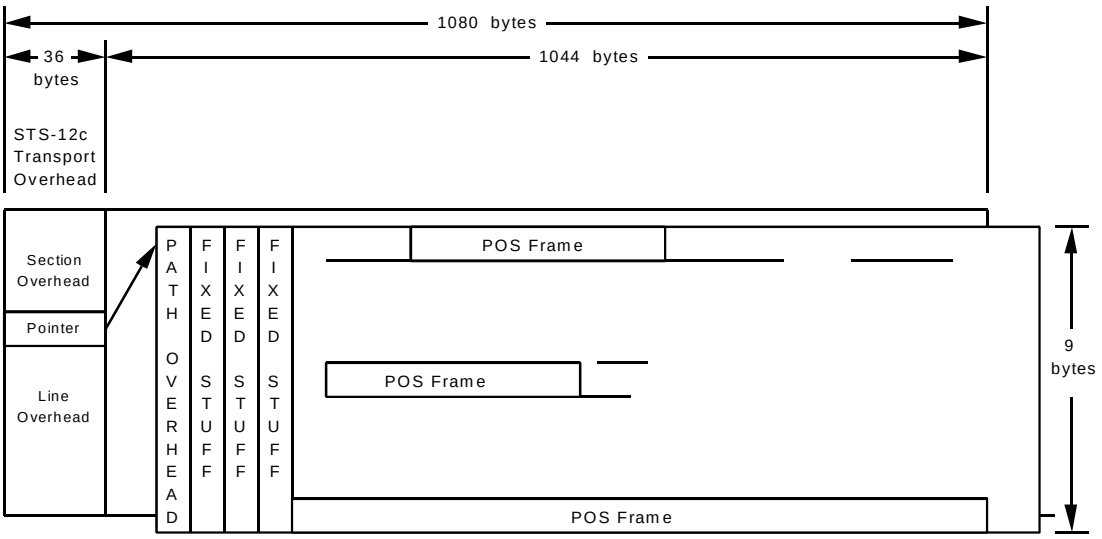

Figure 2. STS-12c (STM-4) Mapping of POS Frames

2.2 IP over SONET

In an IP (through PPP) over SONET infrastructure, POS links provide high bandwidth pipes that can be used to interconnect high-speed routers. A typical IP over SONET network configuration is shown in Figure 3.

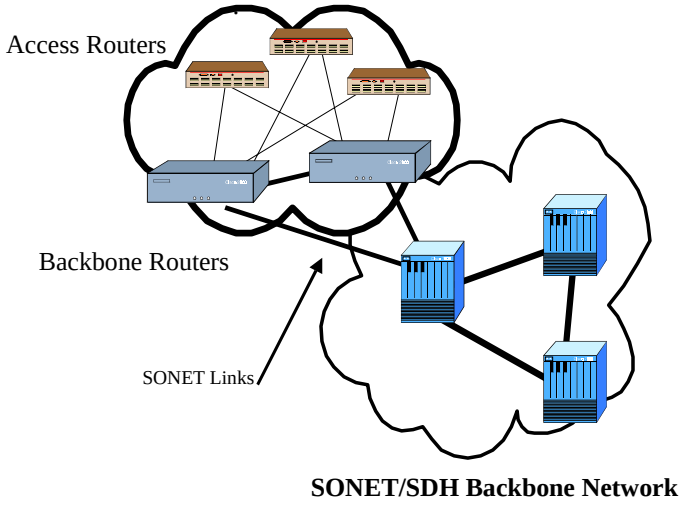

Figure 3. IP over SONET Internet Backbone

PPP is the Internet standard for transporting multi-protocol IP packets over serial links. The PPP mapping into SONET/SDH is done in accordance with RFC 1619*, "PPP over SONET/SDH,"* and RFC 1662, *"PPP in HDLC-like Framing,"* at OC-3/STM-1, OC-12/STM-4 and OC-48/STM-16 rates. PPP is described specifically in RFC 1661 *"The Point-to-Point Protocol (PPP),"* and provides multi-protocol encapsulation, error control, and link initialization control features.

PPP provides a standard method for transporting multi-protocol datagrams[2] over point-to-point links. These links provide full-duplex simultaneous bi-directional operations, and are assumed to deliver packets in order. PPP is responsible for encapsulating the multi-protocol datagrams and implements both link control and network control protocols. In order to delineate the packets in the SONET SPE, some form of framing is required. RFC 1662 describes an HDLC-like framing structure for PPP encapsulated packets as shown in Figure 4. Delineation of PPP encapsulated IP datagrams is performed using Flag Sequence recognition and byte stuffing/de-stuffing techniques described in Section 3.3.2.

Flag 0x7E	Address 0xFF	Control 0x03	Protocol 8/16 bits	Information	Padding	FCS 16/32	Flag 0x7E

Figure 4. PPP HDLC-like Frame

The Protocol Stack for mapping IP over SONET using PPP is shown in Figure 5. IP datagrams are encapsulated into PPP packets, which are then framed into POS Frames using HDLC-like framing according to RFC 1662, and finally, mapped byte synchronously into the SONET SPE.[3] Architecturally, IP over SONET constitutes a single logical connection using a single physical connection via router interconnects.

[2] A datagram is the unit of transmission in the network layer (such as IP). A datagram is encapsulated in one or more packets passed to the data link layer.
[3] Note that scrambling must be performed immediately before inserting the POS frames into the SONET SPE.

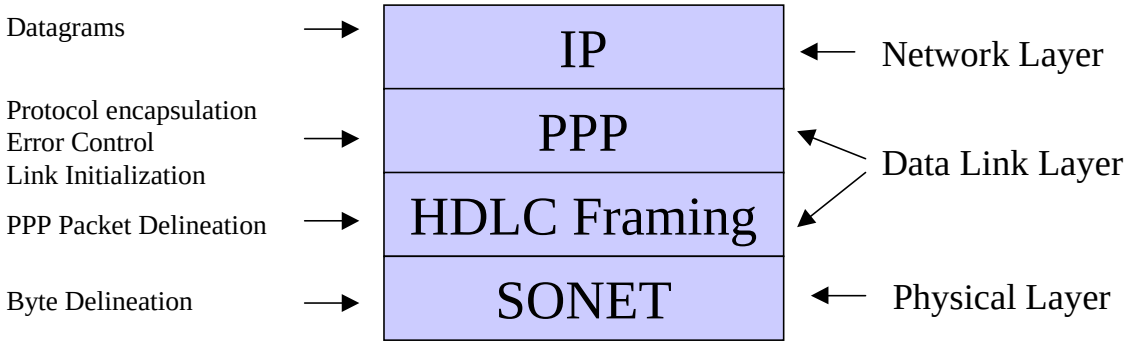

Figure 5. IP over SONET Protocol Stack

2.3 Frame Relay over SONET

Typically, a user is connected to a frame relay network through a router or other Frame Relay Access Device (FRAD). The router communicates with the Frame Relay switch via a Frame Relay User Network Interface. Frame Relay is considerably more efficient than previous data networks such as X.25 as expensive error correction protocols are eliminated. Frame Relay uses variable length frames (Protocol Data Units) which allow it to support the internetworking of various types of networks. However, variable length frames mean variable delay, which poses difficulties for transmitting delay sensitive data (i.e. voice, video). Several solutions are currently under debate to address this issue.

Frame Relay is a layer 2 protocol, and its framing structure is derived from HDLC and the HDLC derivative LAPD, called LAPF, as shown in Figure 6. Unlike HDLC, the control and address fields are combined into one field in the Frame Relay header, called the Data Link Channel Identifier (DLCI). The Frame Relay system supports virtual circuit multiplexing and demultiplexing through the use of the DLCI field. The payload fields of the frames on the channel may contain traffic from multiple users, each payload field identified with a unique DLCI. This 10 bit number[4] is identical to a virtual circuit number in a network layer protocol. Essentially, Frame Relay is eliminating much of the functions of the network layer to perform layer 2 switching.

Figure 6. The Frame Relay Frame

[4] The 10 bit number can be expanded with an extended address option

Frame Relay has faster transmission rates than older technologies with transmission rates currently defined up to the DS-3 (45 Mbit/s) line rate. There is no standard for Frame Relay beyond this data rate, and the Frame Relay Forum is not working on any. It is, however, possible to map Frame Relay Protocol Data Units (PDUs) directly into the SONET SPE, and achieve OC-3 data rates and above. This data rate increase might not be achieved seamlessly and will require a standardization effort. Putting these issues aside, a typical Frame Relay over SONET network is illustrated in Figure 7.

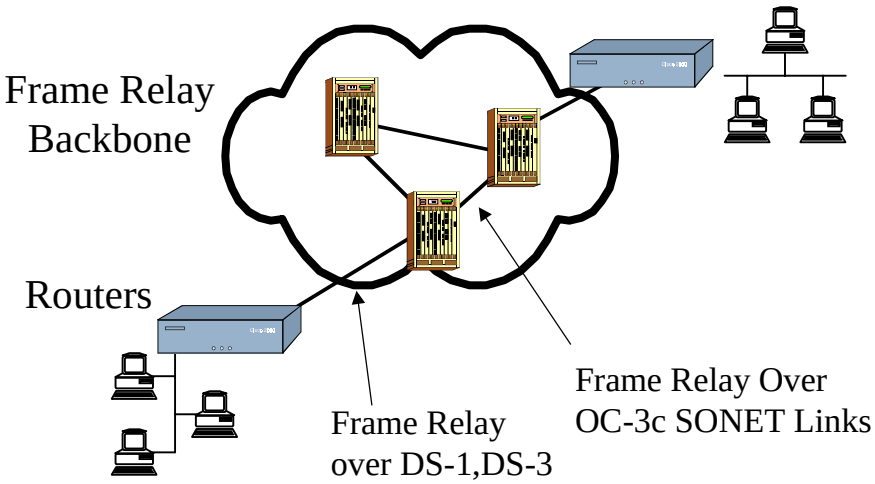

Figure 7. Frame Relay over SONET Network

As with PPP over SONET, a method is needed to byte synchronously delineate the Frame Relay packets prior to insertion into the SONET SPE. Inserting the Frame Relay packets into an HDLC frame as described in RFC 1662 is one approach. The protocol stack for this approach is illustrated in Figure 8.

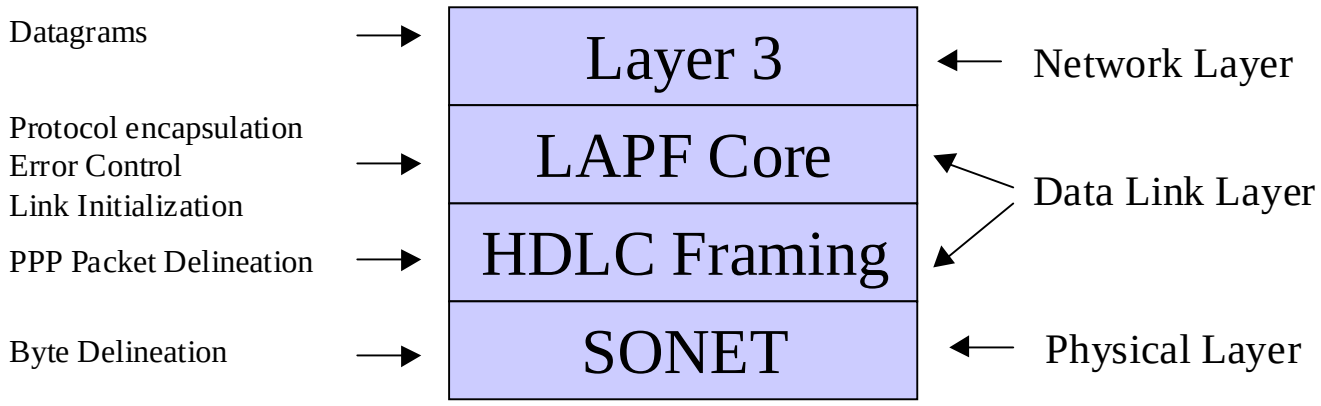

Figure 8. Frame Relay over SONET Protocol Stack

2.4 Ethernet over SONET

Ethernet is another Layer 2 Protocol that has the potential to be mapped directly into the SONET SPE. Applications can be found in large corporate networks, providing a SONET fat pipe between backbone switches in both LAN and MAN applications. A possible Ethernet over SONET network is shown in Figure 9.

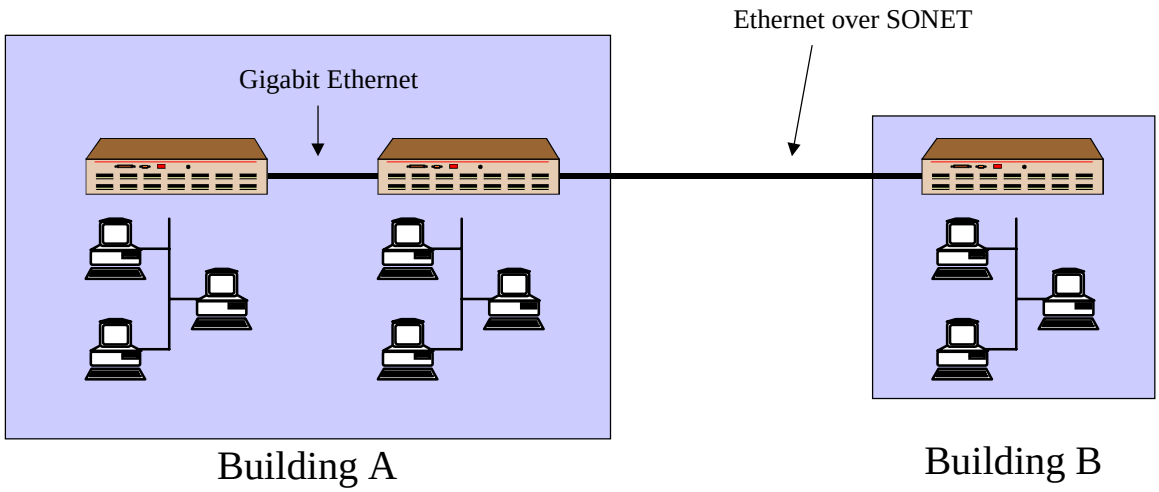

Figure 9. Ethernet over SONET Network

As with PPP and Frame Relay over SONET, a method is needed to byte synchronously delineate the Ethernet packets prior to insertion into the SONET SPE. Inserting the Frame Relay packets into an HDLC frame as described in RFC 1662 is one approach. The corresponding Ethernet over SONET Protocol stack is illustrated in Figure 10.

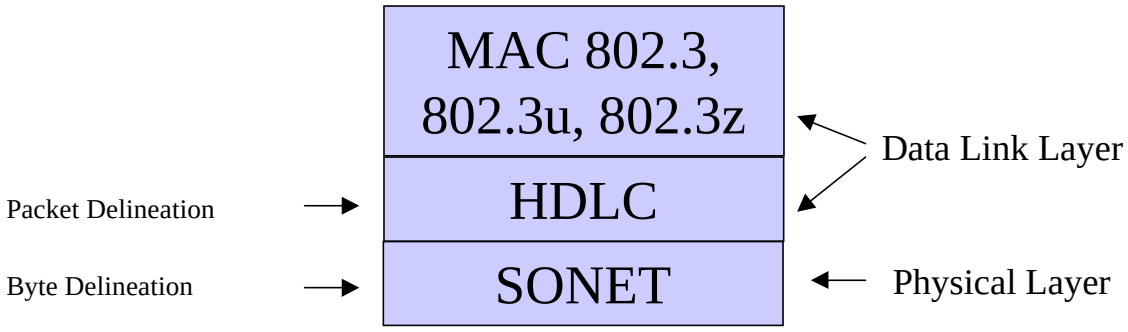

Figure 10. Ethernet over SONET Protocol Stack

Another approach is to add a preamble and frame length field to the beginning of the frame, together with a checksum at the end of the frame, as shown in Figure 11. This entire encapsulated frame could be inserted into the SONET SPE with idle flags inserted when no data is being sent. The preamble would be used to delineate the Ethernet Frames, and the length field would be used to determine end of frame. In this scheme there is no need for byte stuffing/destuffing as flag patterns are not searched for during packet transmission or reception. A draw back with the scheme is that the entire Ethernet packet (potentially greater than 1500 bytes) would have to be buffered prior to transmission so that the length of the frame can be calculated.

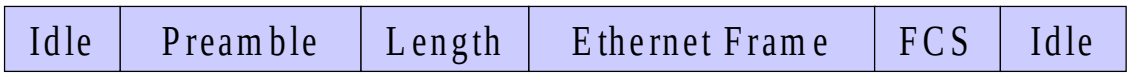

Figure 11. Encapsulated Ethernet Frame

3. The Physical Layer Device

The functions of HDLC processing and SONET mapping logically fall into the responsibility of the Physical Layer Device (PHY). A POS PHY device could look something like that shown in Figure 12.

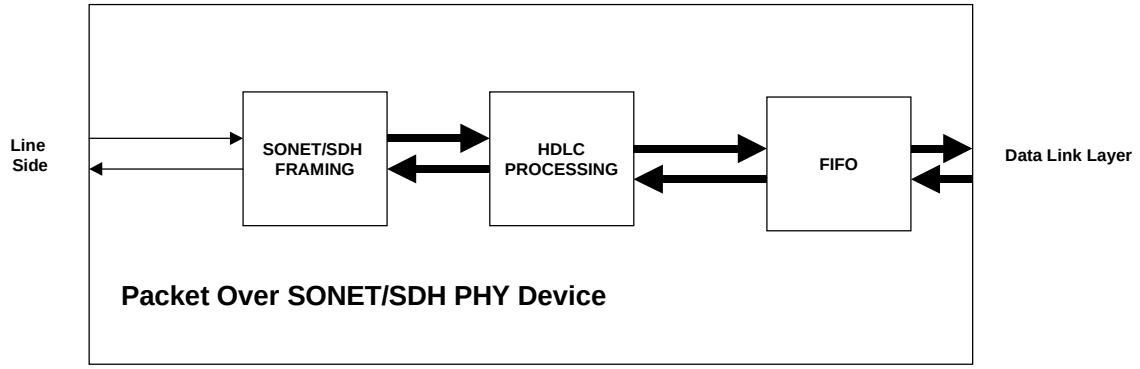

Figure 12. Packet over SONET PHY Device

3.1 Framing

SONET/SDH Framing is defined clearly by Bellcore, ANSI and ITU and there are no issues regarding its implementation. The only overhead byte usage that is affected by the payload type is the path signal label code C2. It is proposed by RFC 1619 that this code be set to cf (207d) to identify a POS payload. Since RFC 1619 does not include data scrambling, and given that this feature has been demonstrated essential to the safe operation of POS links, some are proposing to define a new value for the C2 byte which would be used to identify POS links built with a data scrambler.

3.2 Data Scrambling

Data scrambling provides for a more robust system preventing the injection of hostile patterns (killer packets) into the data stream. Although explicitly mentioned as not being required in RFC 1619, it is industry consensus that a payload scrambler is necessary to provide reliable network operations.

3.2.1 The Necessity for Scrambling

SONET utilizes a frame synchronous scrambler to ensure an adequate number of data transitions for clock recovery purposes. The SONET generating polynomial is $X^7 + X^6 + 1$ and the sequence length is 127. Note that the framing bytes (A1 and A2) and the trace/growth bytes (J0/Z0) are not scrambled. The SONET scrambler was deemed sufficient to provide payload transparency for a multiplexed payload (as in the case with virtual tributaries coming from different sources). However, if a user can gain access to a significant portion of the SPE (as with ATM or IP), the SONET scrambler no longer provides sufficient payload transparency.

Essentially, the SONET scrambler is reset each frame by setting each of the registers to all ones on the most significant bit of the byte following the STS-1 number N J0/Z0 byte. A seven stage shift register, then, produces a pseudo random sequence which is XOR'ed with the incoming data. It is possible for a user to obtain the output from the shift registers and transmit this pattern repetitively (such as in an IP datagram). The user has a 1/127 chance of aligning this pattern with that coming from the scrambler. By continuously repeating this pattern, it can eventually align with the scrambler output and produce an all zeros pattern. Loss of Signal, Loss of Framing and De-synchronization may occur, resulting in system failure.

Currently a number of scramblers have been proposed to ensure adequate payload transparency.

- $x^{43}+1$ (self-synchronous)
- $1+x^{2}+x^{19}+x^{21}+x^{40}$ (Frame Synchronous)
- $x^{43}+x^{23}+1$ (self-synchronous)

3.2.2 The $x^{43}+1$ Self-Synchronous Data Scrambler

The $x^{43}+1$ self-synchronous data scrambler is used with ATM SONET mapping and is the leading candidate to be used with Packet SONET mapping. A self-synchronous data scrambler has the merit of not requiring any SONET overhead bytes to operate, providing a simple integration into the current set of standards. The scrambler is shown in Figure 13, and the descrambler is shown in Figure 14.

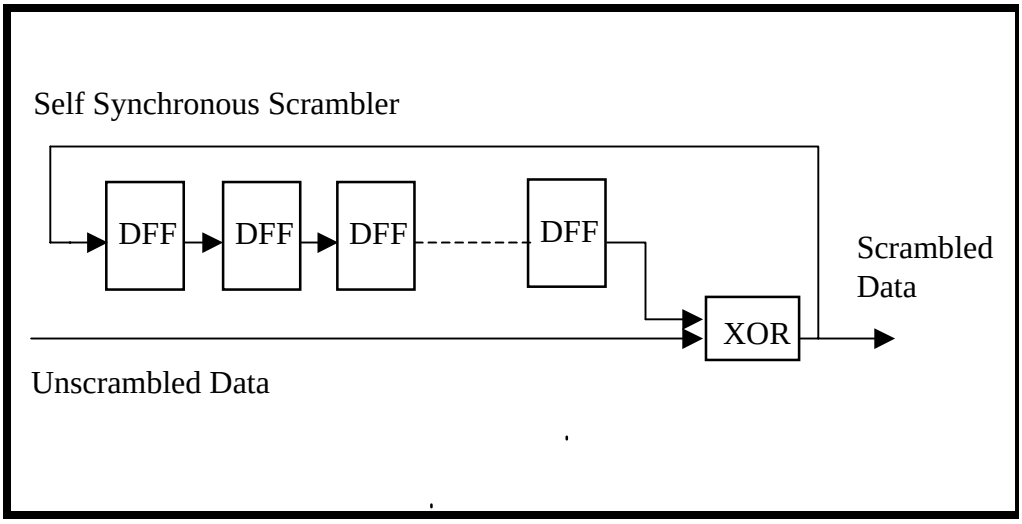

Figure 13. Self-Synchronous Data Scrambler

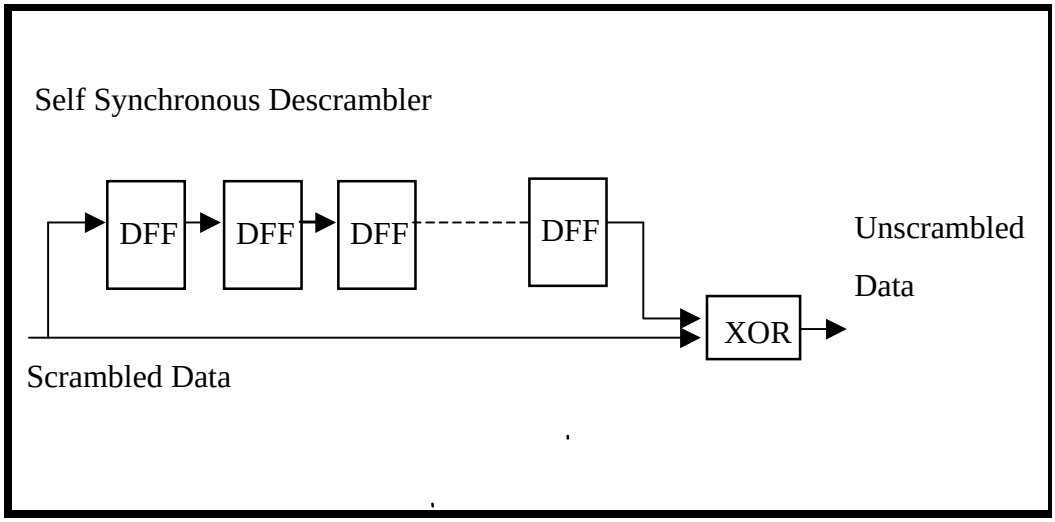

Figure 14. Self-Synchronous Data Descrambler

The scrambler should scramble the whole packet data, including the FCS and the flags. This scrambler is straight-forward to implement and current Packet-over-SONET implementations use this scrambling method. The scrambler is also shown to be robust for PPP over SONET/SDH applications in the T1X1.5 contribution, "Scramblers for PPP over SONET/SDH: Considerations and Analysis."

3.2.3 The $1+x^2+x^{19}+x^{21}+x^{40}$ Frame Synchronous Scrambler

ANSI submission T1X1.5/97-105 recommends a frame synchronous scrambling methodology. The authors of this submission have since submitted a contribution to T1X1.5 in support of $X^{43}+1$ titled "Scramblers for PPP over SONET/SDH: Consideration and Analysis." However, for completeness, the earlier submission is described.

The scrambler has a polynomial degree of 40, which is sufficient for speeds exceeding OC-192. However, in order to synchronize the scrambler to the descrambler, the scrambler state is transmitted in the SONET Path Overhead bytes. As the scrambler is 40 bits long, 5 bytes are needed to transmit this state. It is proposed that this be accomplished by carrying the state in the H4, Z3 and Z4 bytes over multiple frames.

Although this method seems to provide sufficient payload transparency, the requirement to manipulate SONET bytes prevents backwards compatibility and has serious implications on SONET equipment. Many SONET Physical Layer Implementations do not currently extract or process all the SONET bytes, and an immediate requirement as such will place a significant strain on network providers and equipment manufacturers. Furthermore, SONET and SDH are standardized by multiple organizations (including

Bellcore, ANSI and the ITU-T), and standardizing modifications to overhead byte usage will be a timely process and likely present conformity and compatibility issues.

3.2.4 The $x^{43}+x^{23}+1$ Self-Synchronous Data Scrambler

IETF Internet Draft, "draft-ferguson-pppsonet-selfsynch-00.txt," submitted in November 1997, proposes the self-synchronous data scrambler $x^{43}+x^{23}+1$. It is proposed that the $x^{43}+1$ scrambler has negative interactions with the 16-bit Frame Check Sequence, and that adding a third term to the generator polynomial may improve its behavior.

The submission describes that the only major defect related to the use of self-synchronous scramblers (in particular $x^{43}+1$) with PPP over SONET/SDH is the effect that error multiplication may have on CRC error detection. It points out that the 16-bit FCS is weaker than the 32-bit FCS and although RFC 1619 recommends the use of the 32-bit FCS, strong reasons exists why 16-bit FCS should be supported. These include (1) there are existing POS implementations with the 16-bit FCS and (2) the 16-bit FCS provides a reduced per-frame overhead (perhaps a reasonable tradeoff given the very low error rates provided by SONET). The authors suggest that adding a third term to the scrambler (X^{23}, with 23 chosen arbitrarily) would not negatively impact the 16-Bit FCS' ability to detect errors. The submission also notes that $x^{43}+x^{23}+1$ scrambler has no benefit over the $x^{43}+1$ scrambler on 32-bit FCS error detection.

As noted in T1X1.5 contribution, "Scramblers for PPP over SONET/SDH: Considerations and Analysis," the emphasis in IP over SONET implementations is on data transport, with errored packets discarded by the HDLC FCS check. Furthermore, additional checks exist at the IP and TCP layers. The contribution goes on further to demonstrate the robustness of the $x^{43}+1$ scrambler and the negligible probability of successfully guessing the state of the scrambler. As long as equipment can handle a relatively low transition density in 43 bit periodic sequences, malicious attacks will not be successful.

3.3 HDLC Processing

For PPP over SONET/SDH applications, the mapping function is defined by RFC-1619, with the addition of the self-synchronous scrambler.

Framing for octet-synchronous links is defined by RFC-1662. The Byte Serial HDLC Functionality is shown in Figure 15.

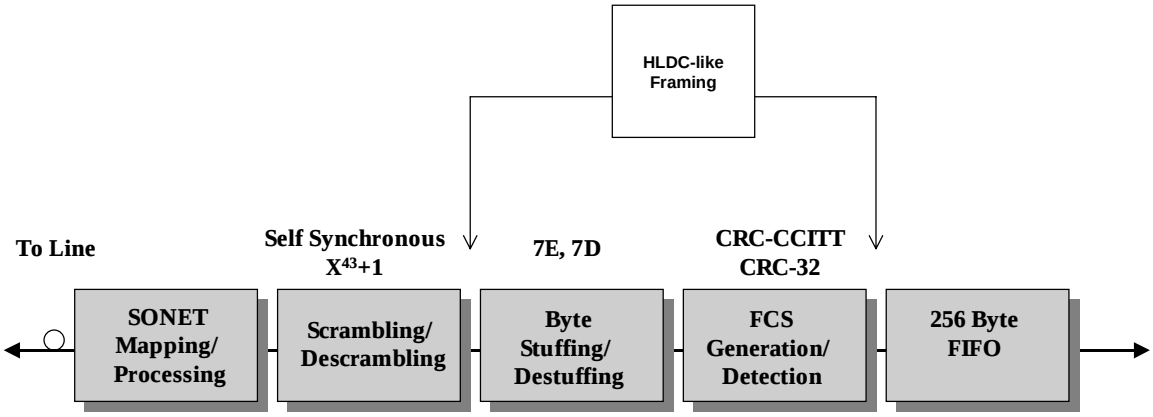

Figure 15. Byte Serial HDLC Functionality

3.3.1 Flag Sequence

Octet synchronous HDLC-like framing must be performed to encapsulate the PPP datagrams prior to mapping into the SONET/SDH payload. Octet synchronous HDLC differs from bit synchronous HDLC in that byte-by-byte encapsulation is performed rather than a bit-by-bit encapsulation. Basically, the flag character, the escape character and possibly control characters are escaped by pre-pending the escape character and XORing them with 0x20. Figure 16 shows the POS frame including payload, FCS and the framing flags.

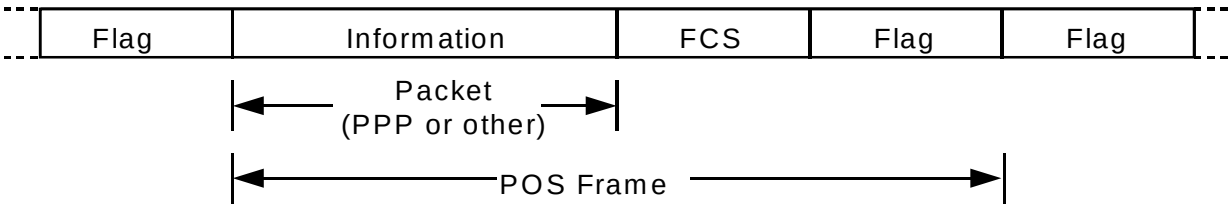

Figure 16. Packet over SONET Frame Format

The Flag Sequence indicates the beginning and end of frame. The octet stream is examined on an octet-by-octet basis for the value 01111110 (hexadecimal 0x7e).

3.3.2 Transparency

An octet stuffing procedure is used for transparency. The Control Escape octet is defined as binary 01111101 (hexadecimal 0x7d), most significant bit first.

After FCS computation, the transmitter examines the entire frame between the two Flag Sequences. Every Flag Sequence and Control Escape Sequence contained within the user data is replaced by a two octet sequence consisting of the Control Escape octet followed by the original octet exclusive-or'd with hexadecimal 0x20, as illustrated in Figure 17.

Original	Escaped
7E (Flag Sequence)	7D-5E
7D (Control Escape)	7D-5D

Figure 17. Byte Stuffing Escape Codes

On the receiving side, prior to FCS computation, each Control Escape octet is removed and the following octet is exclusive-or'd with hexadecimal 0x20, unless it is the Flag Sequence. The Control Escape character followed by the Flag Sequence character has a special meaning. This is the abort sequence, which can be used to abort a frame. A transmitter can append it to a frame to cause the receiver to discard this particular frame. The other widely used technique for the transmitter to abort a frame is to invert the FCS bytes. Although providing the same results, the former technique is preferred over the latter for performance monitoring reasons.

3.3.3 FCS Generator

The FCS Generator performs a CRC-CCITT or CRC-32 calculation on the whole POS frame, before byte stuffing and data scrambling. The FCS generator is illustrated in Figure 18. A parallel implementation of the CRC polynomial is used. The CRC algorithm for the frame checking sequence (FCS) field is either a CRC-CCITT or CRC-32 function. The CRC-CCITT is two bytes in size and has a generating polynomial $g(X) = 1 + X^5 + X^{12} + X^{16}$. The CRC-32 is four bytes in size and has a generating polynomial $g(X) = 1 + X + X^2 + X^4 + X^5 + X^7 + X^8 + X^{10} + X^{11} + X^{12} + X^{16} + X^{22} + X^{23} + X^{26} + X^{32}$. The first FCS bit transmitted is the coefficient of the highest term. When transmitting a packet from the Transmit FIFO, the FCS Generator appends the result after the last data byte, before the closing flag.

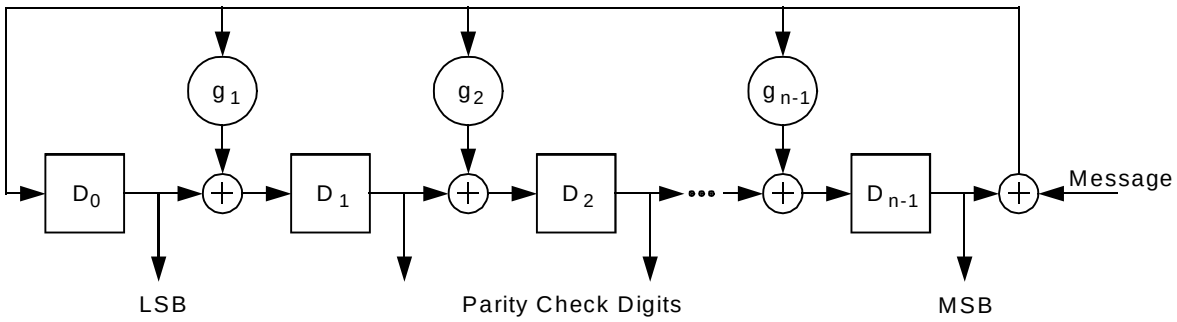

Figure 18. CRC Generator

3.4 POS-PHY™ FIFO Interface

The PHY should provide a deep enough FIFO (i.e. 256 bytes) to separate the line timing from the link layer system timing, and to handle timing differences caused by the removal of escape characters.

The POS PHY™ interface is used for the interconnection of Physical Layer (PHY) devices to Link Layer devices implementing POS. POS-PHY™ fulfills the need of system designers to target a standard POS Physical Layer interface.

POS-PHY™ is being developed with the cooperation of the SATURN™ Development Group. POS-PHY™ Level 2 covers application bit rates up to and including 622 Mbit/s. POS-PHY™ Level 3, will provide an extension to 2488 Mbit/s. POS-PHY™ defines the requirements for interoperable single-PHY (one PHY layer device connects to one Link Layer device) and multi-PHY (several PHY layer devices connect to one Link Layer device) applications. It stresses simplicity of operation to allow forward migration to more elaborate PHY and Link Layer devices.

A complete specification for POS-PHY™ Level 2 is available from PMC-Sierra to provide a reference to independent developers of integrated circuits or system-level circuits who wish to interoperate with SATURN Compatible components. The specification is document PMC-971147, "*POS-PHY™ SATURN COMPATIBLE PACKET OVER SONET INTERFACE SPECIFICATION FOR PHYSICAL LAYER DEVICE (Level 2)*". POS-PHY™ Level 3 is under development and is not available at the time this document was written. Please contact PMC-Sierra's marketing department for the status of its development.

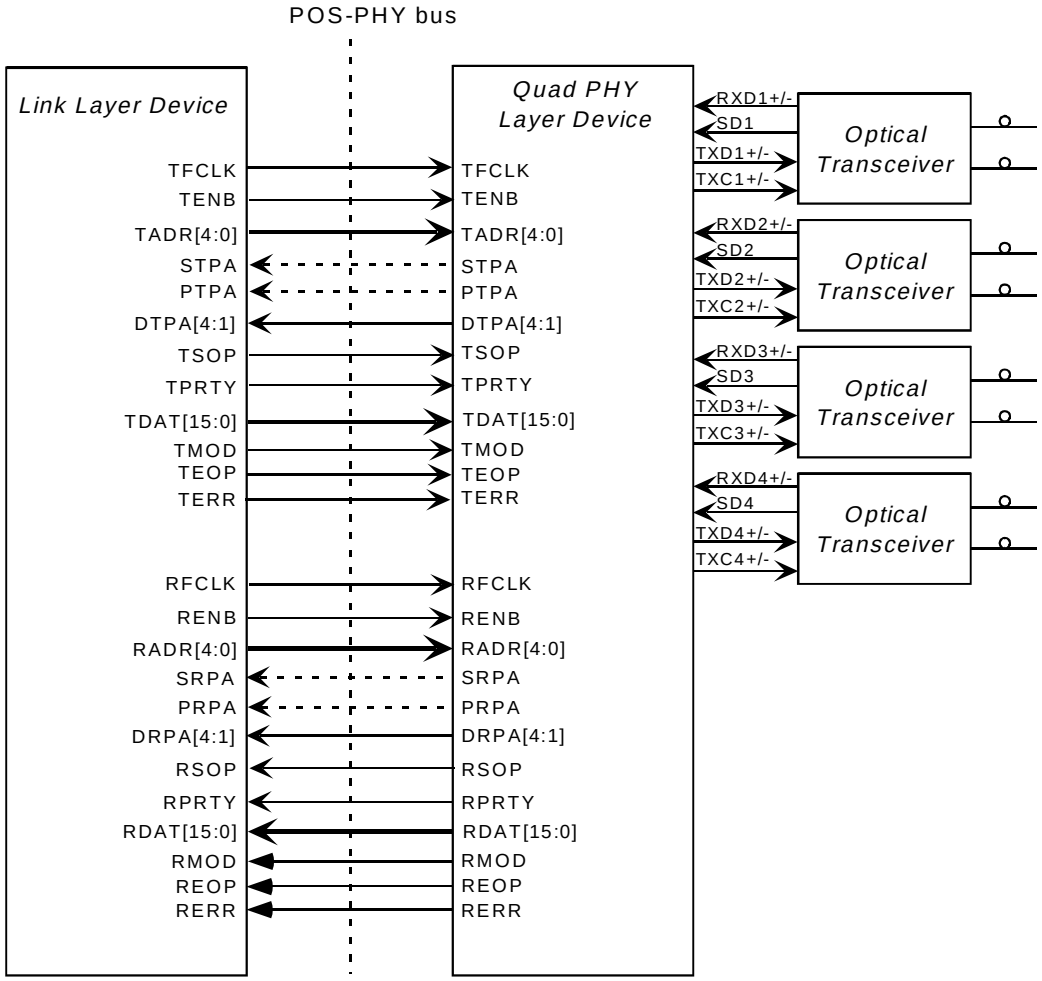

Figure 19. Example of a POS-PHY™ Physical to Link Layer Interface

4. Conclusion

Packet-Over-SONET/SDH (POS) is an emerging technology for carrying IP and other data traffic over the SONET/SDH backbone. This paper has shown how this technology can be used to carry various traffic types over a SONET link, and covers the various aspects of its physical implementation. There is currently an industry consensus around most implementation requirements. The choice of a data scrambling technique as a mean of increasing the reliability of POS links is actively being investigated and a consensus should emerge shortly. Packet over SONET has the potential to play an important role in the deployment of ever faster data communication links.