

## PPP Serial Data Transport Protocol (SDTP)

### Status of This Memo

This memo provides information for the Internet community. This memo does not specify an Internet standard of any kind. Distribution of this memo is unlimited.

### Abstract

The Point-to-Point Protocol (PPP) [1] provides a standard method for transporting multi-protocol datagrams over point-to-point links. PPP defines an extensible Link Control Protocol, and proposes a family of Network Control Protocols for establishing and configuring different network-layer protocols.

This document describes a new Network level protocol (from the PPP point of view), PPP Serial Data Transport Protocol, that provides encapsulation and an associated control protocol for transporting serial data streams over a PPP link. This protocol was developed for the purpose of using PPP's many features to provide a standard method for synchronous data compression. The encapsulation uses a header structure based on that of the ITU-T Recommendation V.120 [2].

### Table of Contents

1.	Introduction .....	2
2.	SDTP Packets .....	3
2.1	Padding .....	4
2.2	Packet Formats .....	4
3.	Serial Data Control Protocol .....	11
4.	SDCP Configuration Option Format .....	12
4.1	Packet-Format .....	13
4.2	Header-Type .....	13
4.3	Length-Field-Present .....	14
4.4	Multi-Port .....	14
4.5	Transport-Mode .....	15
4.6	Maximum-Frame-Size .....	16
4.7	Allow-Odd-Frames .....	16
4.8	FCS-Type .....	17
4.9	Flow-Expiration-Time .....	18
	SECURITY CONSIDERATIONS .....	19

REFERENCES .....	19
CHAIR'S ADDRESS .....	20
AUTHORS' ADDRESSES .....	20

## 1. Introduction

This document is a product of the TR30.1 ad hoc committee on compression of synchronous data. It represents a component of a proposal to use PPP to provide compression of synchronous data in DSU/CSUs.

In addition to providing support for multi-protocol datagrams, the Point-to-Point Protocol (PPP) [1] has defined an effective and robust negotiating mechanism that can be used on point to point links. When used in conjunction with the PPP Compression Control Protocol [3] and one of the PPP Compression Protocols [4-10], PPP provides an interoperable method of employing data compression on a point-to-point link.

This document provides a PPP encapsulation for serial data, specifying a transport protocol, PPP Serial Data Transport Protocol (PPP-SDTP), and an associated control protocol, PPP Serial Data Control Protocol (PPP-SDCP). When these protocols are added to above mentioned PPP protocols, PPP can be used to provide compression of serial data on a point-to-point link.

This first edition of PPP-SDTP/SDCP covers HDLC-like synchronous serial data and asynchronous serial data. It does this by using a terminal adaption header based on that of ITU-T Recommendation V.120 [2]. Support may be added in the future for other synchronous protocols as the marketplace demands.

The V.120 terminal adaption header allows transported data frames to be split over several packets, supports the transport of DTE port idle and error information, and optionally supports the transport of DTE control state information.

In addition to the V.120 Header, fields can be added to the packet format through negotiation to provide support for features not included in the V.120 header. The extra fields are: a Length Field, which is used to distinguish packets in compound frames, and a Port field, which is used to provide multi-port multiplexing capability. The protocol also allows reserved bits in the V.120 header to be used to transport non-octet aligned frames and to provide a flow control mechanism.

To provide these features, PPP-SDTP permits a single frame format to be selected from several possible formats by using PPP-SDCP negotiation. The terminal adaption header can be either fixed length or variable length, to allow either simplicity or flexibility.

The default frame format places the terminal adaption header at the end of the packet. This permits optimal transmitter timelines when user frames are segmented and compression is also used in conjunction with this protocol.

## 2. SDTP Packets

Before any SDTP packets may be communicated, PPP must reach the Network-Layer Protocol phase, and the SDTP Control Protocol must reach the Opened state.

By default, exactly one SDTP packet is encapsulated in the PPP Information field, where the PPP Protocol field indicates type hex 0049 (PPP-SDTP). If the Length-Field-Present Configuration Option and the LCP Compound-Frames Configuration Option are successfully negotiated, multiple SDTP packets may be placed in the PPP Information field, and they are distinguished by the presence of Length fields in each packet.

The maximum length of the SDTP datagram transmitted over a PPP link is limited only by the negotiated Maximum-Frame-Size and the maximum length of the Information field of a PPP encapsulated packet. Note that if compression is used on the PPP link, this the maximum length of the SDTP datagram may be larger or smaller than the maximum length of the Information field of a PPP encapsulated packet, depending on the particular compression algorithm and protocol used.

ITU-T Recommendation V.120 [2] defines an adaption header that is used with its asynchronous and synchronous modes of operation. SDTP packets include this header as a Header field to provide the protocol adaption function. Using negotiation, additional fields can be added to the packet to provide sequencing and multiplexing capability within SDTP. SDTP also has an option of using the reserved bits of the header to provide a flow control mechanism and support for transporting non-octet aligned data frames.

The default SDTP packet format is designed to allow the efficient use of the protocol's segmentation feature when combined with a PPP Compression Protocol [4-10]. This format is a little different from other PPP NCP's in that data is read from both ends of the packet. The Header field is placed at the end of the SDTP packet, with the order of the octets reversed. This somewhat unique format has been selected to allow optimal transmitter timelines when compression is

used and transported data frames are split into multiple SDTP packets. In such a situation, the Header field contains the information about whether the data is split into multiple packets or not, so if it is located at the end of a packet, the decision can be made after observing the compressed size of the packet. The Header field can then simply be run through the compressor after the decision has been made.

When the Header field is placed before the data, as in the optional packet format, the transmitter must make the decision about whether to split a frame over multiple packets without knowing about the compressibility of the frame. Therefore the optional format is designed to be used when transported frames are not split into multiple SDTP packets or where SDTP is not coupled with compression. It is believed that this format may be useful for some hardware implementations.

## 2.1. Padding

If padding is used, SDTP packets require the use of the Length Field or the previous negotiation of the LCP Self-Describing-Padding Configuration Option [11].

## 2.2. Packet Formats

The default SDTP packet format is shown below. The fields are transmitted from left to right.

```

      0               1               2               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|           PPP Protocol ID           |   Transported Data ...
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| Header - H   |
+---+---+---+---+---+

```

The two complete frame formats are shown below: Header-Last and Header-First. Header-Last is the default packet format. The additional fields provided support for: Control State Information (CS), multiple packets and multi-port multiplexing. Again, the fields are transmitted from left to right. Descriptions of the fields follow the packet formats.

## Header-Last

```

      0                1                2                3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|           PPP Protocol ID           |           (Length)           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|   (Port)   | Transported Data / (Odd-Pad) ... |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Header - (CS) :       H       |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

## Header-First

```

      0                1                2                3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|           PPP Protocol ID           |           (Length)           |
+-----+-----+-----+-----+-----+-----+-----+-----+
|   (Port)   | Header - H :       (CS)       |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Transported Data / (Odd-Pad) ... |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

## PPP Protocol ID

The PPP Protocol ID field is described in the Point-to-Point Protocol Encapsulation [1].

When the SDTP Protocol is successfully negotiated by the SDTP Control Protocol (SDCP), the value is 0049 hex. This value may be compressed to one octet when Protocol-Field-Compression is negotiated, or if one of the PPP compression protocols [4-10] is used.

## Length

The optional Length field is present in every SDTP packet upon successful negotiation of the Length-Field-Present Configuration Option.

The value of the Length field is the combined lengths of the Length, Port (if present), Header, Transmitted Data, and Odd-Pad (if present) fields in octets.

The length of the Length field defaults to one octet. Valid lengths are from 2 to 255 octets, since each packet must include

at least a one octet Header field.

If desired, the length field can be negotiated to be two octets in length. In that case, valid lengths are from 2 to 65535 octets, and the field is transmitted most significant octet first.

In either case, a length of 0 means that the combined length is the same as the length of the remainder of the PPP Information Field.

#### Port

The optional Port field is present in every SDTP packet upon successful negotiation of the Multi-Port Option.

The length of the Port field is one octet. Valid Port numbers are 0 to 254. Port number 255 is reserved for control purposes (see section on flow control).

#### Header

The Header field is the terminal adaption header from ITU-T Recommendation V.120. As specified in that document, it contains up to two octets: The terminal adaption header octet (H), and the optional header extension for control state information (CS). SDTP only supports the protocol sensitive operation of V.120; bit transparent operation is not supported. The descriptions of the header bits provided below are derived from the descriptions provided in Recommendation V.120. In addition to the bit definitions of V.120, SDTP optionally permits the use of reserved bits to be used for flow control and to provide support for non-octet aligned frames.

The length of the Header field is either one or two octets, and is determined by the value of the E bit in the first octet. By default, the E-bit must be set in the H octet and the CS octet is not present. A Configuration Option may be negotiated to allow the use of the CS octet, or even to require its presence in every packet.

## H (V.120 Terminal Adaption Header)

The format of the first octet of the Header field is shown below:

0	1	2	3	4	5	6	7
+	+	+	+	+	+	+	+
E	BR	Res	FC	C2	C1	B	F
+	+	+	+	+	+	+	+

E - Extension Bit

The E bit is the extension bit. If set to 0, it indicates that the Control-2 field is present.

BR - Break / HDLC Idle Bit

In asynchronous mode, the BR bit indicates the invocation of the BREAK function by the DTE. A value of 1 indicates BREAK.

In synchronous HDLC mode, the BR bit is used to indicate that DTE port is receiving HDLC idle condition. A value of 1 indicates this idle condition.

Res - Reserved

This bit is reserved and MUST be set to 0. (This is a reserved bit in V.120.)

FC - Flow Control

This bit can be used for flow control of SDTP traffic on the network, for applications which require it. When SDTP is used in conjunction with data compression, flow control may be needed. Reasons for this could be that the DTE port uses an X.21 interface (and therefore does not have independent control of DTE transmit and receive clocks), or simply that the underlying link layer (such as PPP in HDLC-like Framing) does not include a mechanism for network flow control, so some flow control mechanism is needed.

This bit set to a value of 0 indicates that the receiver is ready to receive data (Flow-On). A value of 1 indicates that the receiver does not wish to receive data and the transmitting peer should stop sending it (Flow-Off). Flow

control operates on a per port basis. Flow control messages on Port 255 affect all ports.

To ensure that a missed Flow-On message cannot cause a hangup condition, a Flow-Off is defined to expire after a time of T1 seconds. If a unit desires to keep its peer in the Flow-Off state for more than T1 seconds, it MUST transmit another Flow-Off message after every period of T1 seconds. A unit that receives a Flow-Off message may resume transmitting T1 seconds after the last Flow-Off was received. The value of T1 is controlled by the Flow-Expiration-Time Configuration Option. The default value is 10 seconds. There is not a separate value for T1 for each port; all ports use the same T1 value.

(This bit is a reserved bit in V.120, which requires the bit to be set to a value of zero. The above definition of flow control provides compatibility with this definition when flow control is not used.)

#### C1, C2 - Error Control Bits

The C1 and C2 bits are used for DTE port Error detection and transmission. Their meaning is defined in the following table:

		Meaning	
C1	C2	Synchronous	Asynchronous
0	0	No Error Detected	No Error Detected
0	1	FCS Error (DTE)	Stop-bit Error
1	0	Abort	Parity Error on the Last Character in Frame
1	1	DTE Overrun*	Stop-bit and Parity Error



Appropriate responses to these bits are provided in Sections 2.2.1 and 2.2.2 of the V.120 standard (where R reference point is translated to mean DTE port.)

#### B, F - Segmentation Bits

The B and F bits are used for segmenting and reassembly of the transported frames in synchronous HDLC mode. Setting the B bit to 1 indicates that the packet contains the beginning of a transported frame or a Begin Frame. Setting the F bit indicates that the packet contains the final portion of a transported frame, or a Final Frame. A packet that contains neither the beginning of a frame nor the end is said to contain a Middle Frame. For asynchronous mode and bit transparent mode operation both bits MUST be set to 1. The following table summarizes the use of these bits:

Application			
B	F	Synchronous	Asynchronous
1	0	Begin Frame	Not Applicable
0	0	Middle Frame	Not Applicable
1	0	Final Frame	Not Applicable
1	1	Single Frame	Required

#### CS (V.120 optional Header Extension for Control State Information)

The format of the second Header octet (CS) is shown below:

0	1	2	3	4	5	6	7
E	DR	SR	RR	Res	(Odd-Pad Length)		

#### E - Extension Bit

The E bit is the extension bit, and allows further extension of the Header field. It is set to 1, to indicate no further extension of the Header field.

**DR - Data Ready**

This bit set to 1 indicates that the DTE port is activated.

**SR - Send Ready**

This bit set to 1 indicates that the DTE is ready to send data.

**RR - Receive Ready**

This bit set to 1 indicates that the DTE is ready to receive data. It can be used for DTE flow control in half-duplex transmissions.

**Res - Reserved**

This bit is reserved and set to 0. (This is a V.120 reserved bit.)

**Odd-Pad Length (Optional)**

The Odd-Pad Length field is used when non-octet aligned HDLC frames are allowed. It is a 3-bit field, that can take on the values of 0 through 7. Its value is the length of the Odd-Pad field in bits. This value is determined as the number of bits necessary to have the combined length of the Transported Data Field and the Odd-Pad Field be aligned with an octet boundary.

If non-octet aligned frames are not allowed, this field is not used and all bits are set to the value of 0. (These bits are reserved in V.120.)

**Transported Data**

The transported data field contains the transported serial data.

When the serial data type has been negotiated to be HDLC-like synchronous, this field will contain all or part of a transported HDLC-like frame.

A sample transported HDLC frame is shown below. The figure does not show bits inserted for transparency.

```

      0               1               2               3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
| Flag:01111110 | (Address, Control and Information Fields) ...
+-----+-----+-----+-----+-----+-----+-----+-----+
|               (FCS)               |
+-----+-----+-----+-----+-----+-----+-----+-----+
| Flag:01111110 |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Only the data between the flags is transported. The flags are not transported. The FCS is transported unless the FCS-Mode Configuration Option has been successfully negotiated otherwise.

#### Odd-Pad

The optional Odd-Pad (Odd Frame Pad) field is used when the transported data frame is non-octet aligned, and the Allow-Odd-Frames Option has been successfully negotiated. It contains the bits that are required to pad the Transported Data field out to an octet boundary. The Odd-Pad field is in the high order bits of the last octet of the Transported Data field. The values of these bits are all zero.

### 3. Serial Data Control Protocol

The Serial Data Control Protocol (SDCP) is responsible for configuring, enabling and disabling the SDTP modules on both ends of the point-to-point link. SDCP uses the same packet exchange mechanism and state machine as the Link Control Protocol. SDCP packets may not be exchanged until PPP has reached the Network-Layer Protocol phase. SDCP packets received before this phase is reached SHOULD be silently discarded.

The Serial Data Control Protocol is exactly the same as the Link Control Protocol [1] with the following exceptions:

#### Frame Modifications

The packet may utilize any modifications to the basic frame format which have been negotiated during the Link Establishment phase.

#### Data Link Layer Protocol Field

Exactly one SDCP packet is encapsulated in the PPP Information field, where the PPP Protocol field indicates type hex 8049 (PPP-SDCP).

### Code Field

Only Codes 1 through 7 (Configure-Request, Configure-Ack, Configure-Nak, Configure-Reject, Terminate-Request, Terminate-Ack, and Code-Reject) are used. other Codes SHOULD be treated as unrecognized and SHOULD result in Code-Rejects.

### Timeouts

SDCP packets may not be exchanged until PPP has reached the Network-Layer Protocol phase. An implementation SHOULD be prepared to wait for Authentication and Link Quality Determination to finish before timing out waiting for a Configure-Ack or other response. It is suggested that an implementation give up only after user intervention or a configurable amount of time.

### Configuration Option Types

SDCP has a distinct set of Configuration Options which are defined in this document.

## 4. SDCP Configuration Option Format

SDCP Configuration Options allow modifications to the default SDCP characteristics to be negotiated. If a Configuration Option is not included in a Configure-Request packet, the default value for that Configuration Option is assumed.

SDCP uses the same Configuration Option format defined in LCP [1], with a separate set of Options.

The Option Types are:

- 1 Packet-Format
- 2 Header-Type
- 3 Length-Field-Present
- 4 Multi-Port
- 5 Transport-Mode
- 6 Maximum-Frame-Size
- 7 Allow-Odd-Frames
- 8 FCS-Type
- 9 Flow-Expiration-Time

Note that Option Types 5-8 are specific to a single port and require port numbers in their format. Option Types 6-8 are specific to the HDLC-Synchronous Transport-Mode.

#### 4.1. Packet-Format

This option selects whether the Header field precedes or follows the data field. When the Header field follows the data field, the order of its octets are reversed.

```

      0                               1                               2
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-----+-----+-----+-----+-----+-----+-----+-----+
|      Type      |      Length      |      Format      |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Type

1

Length

3

Format

```

    0  Header-Last  (default)
    1  Header-First

```

#### 4.2. Header-Type

This option selects the type of the Header field. The Header-Type of H-and-CS means that the CS octet will be present if indicated by the E-bit in the H-octet. The Header-Type of H-and-CS-Always signifies that both the H and CS octets are present in every packet.

```

      0                               1                               2
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-----+-----+-----+-----+-----+-----+-----+-----+
|      Type      |      Length      | Header-Type  |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Type

2

Length

3

## Header-Type

- ```
0    H-Only (default)
1    H-and-CS
2    H-and-CS-Always
```

### 4.3. Length-Field-Present

By default, a PPP Information Field contains only a single SDTP packet, and an SDTP Packet does not contain a length field. Successful negotiation of this option causes all SDTP packets to contain the length field, and allows SDTP packets to be contained in compound frames (see LCP Compound-Frames Configuration Option [11]).

This option is required if the LCP Length-Field-Present Configuration option has been negotiated.

The size of the Length field is negotiated via the Length-Size parameter.

[illegible]

## Type

3

## Length

3

## Length-Size

- ```
0   No Length Field (default)
1   Length field of 1 octet
2   Length field of 2 octets
```

#### 4.4. Multi-Port

By default, packets do not contain a port number and all packets are sent to the default port, Port 0. The Successful negotiation of the Multi-Port configuration option means that every packet will contain a port number. The maximum port number, and hence the number of ports, is negotiated by using the Max-Port-Num field. A value of 0 specifies that a single port is to be used and no port field will be

present in an SDTP packet. (This is the same as not negotiating or rejecting this option.) Port numbers begin with 0 and range to 254. Port number 255 is reserved for control purposes (see section on flow control).

Protocol Specific negotiations which are on a per port basis, require the port number to be specified as part of the configuration negotiation.

```

      0                               1                               2
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-----+-----+-----+-----+-----+-----+-----+-----+
|      Type      |      Length      | Max-Port-Num  |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Type

4

Length

3

Max-Port-Num

The maximum port number that can be used. The number of ports present is Max-Port-Num + 1. The value can range from 0 to 254.

#### 4.5. Transport-Mode

This parameter selects the mode of transport for the specified port.

```

      0                               1                               2                               3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-----+-----+-----+-----+-----+-----+-----+-----+
|      Type      |      Length      |      Port      |      Mode      |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Type

5

Length

4

## Port

The port for which this option applies.

## Mode

The transport mode to be used for this port.

- ```
0    HDLC Synchronous (default)
1    Asynchronous
```

#### 4.6. Maximum-Frame-Size

This parameter specifies the maximum number of octets allowed in a transported data frame.

```

0                                1                                2                                3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+--+---+-----+-----+-----+-----+-----+-----+-----+
|          Type           |      Length       |         Port        |
+-+--+---+-----+-----+-----+-----+-----+-----+-----+
|                                     Maximum-Frame-Size                               |
+-+--+---+-----+-----+-----+-----+-----+-----+-----+

```

## Type

6

## Length

7

## Port

The port for which this option applies.

## Maximum-Frame-Size

The maximum allowed length of a transported data frame in octets. Default is 10,000. Negotiable range is 1 to  $2^{31} - 1$ . The value 0 is reserved to mean no limit. This field is transmitted most significant octet first.

#### 4.7. Allow-Odd-Frames

By default, only octet-aligned data frames are allowed for transport. Successful negotiation of this option allows the transport of non-octet aligned frames. The size of the padding required to align the



frames is carried in the CS Header octet.

Use of Header-Type H-Only is not permitted in conjunction with this option.

```

      0                               1                               2
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Type   |   Length   |   Port   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Type

7

Length

3

Port

The port for which this option applies.

#### 4.8. FCS-Type

By default, the transported data frame FCS is transported. This option allows the FCS to be removed by the transmitter and regenerated by the receiver.

It is important that implementations not use regeneration unless they are using PPP Reliable Transmission [12] or operating over some other layer that will provide reliable notification of a dropped packet. Implementations are not permitted to send an incomplete or bad frame to the user with a good (regenerated) FCS.

This option also selects the type of user FCS that will be regenerated.

```

      0                               1                               2                               3
    0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Type   |   Length   |   Port   |   FCS-Type   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Type

8

Length

4

Port

The port for which this option applies.

FCS-Type

- 0 Transparent-Transport (Default)
- 1 16-bit ITU-T CRC
- 2 32-bit ITU-T CRC

#### 4.9. Flow-Expiration-Time

As described in [section 2.2](#), Flow-Off messages expire after T1 seconds. By default, T1 is 10 seconds. This configuration option allows the value of T1 to be changed.

```

0                               1
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|      Type      |      Length      |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|      Flow-Expiration-Time      |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Type

9

Length

5

Flow-Expiration-Time

The Flow-Expiration-Time field contains a 16 bit unsigned integer which is used to specify the value to be assigned to T1 as follows:  $T1 = \text{Flow-Expiration-Time} / 10 \text{ seconds}$ . Therefore this value is in units of 1/10 of a second, with allowable values from 1 to  $2^{16}-1$  (0.1 to 6553.5 seconds). It is transmitted most significant octet first. The default value is 100 (10 seconds), which all must support.

## Security Considerations

Security issues are not discussed in this memo.

## References

- [1] Simpson, W., ed., "The Point-to-Point Protocol (PPP)", STD 51, [RFC 1661](#), July 1994.
- [2] CCITT Recommendation V.120 (09/92), "Support by an ISDN of Data Terminal Equipment with V-Series Type Interfaces with Provision for Statistical Multiplexing", 1993.
- [3] Rand, D., "The PPP Compression Control Protocol (CCP)", [RFC 1962](#), June 1996.
- [4] Friend, R., and W. Simpson, "PPP Stac LZS Compression Protocol", [RFC 1974](#), August 1996.
- [5] Rand, D., "PPP Predictor Compression Protocol", [RFC 1978](#), August 1996.
- [6] Petty, J., "PPP Hewlett-Packard Packet-by-Packet Compression (HP PPC) Protocol", Work in Progress.
- [7] Carr, D., "[PPP Gandalf FZA Compression Protocol](#)", Work in Progress.
- [8] Schryver, V., "PPP BSD Compression Protocol", [RFC 1977](#), August 1996.
- [9] Schremp, et. al., "PPP Magnalink Variable Resource Compression", [RFC 1975](#), August 1996.
- [10] Schneider, K., "PPP Stacker LZS Compression Protocol using a DCP Header (LZS-DCP)", [RFC 1967](#), August 1996.
- [11] Simpson, W.A., "PPP LCP Extensions", [RFC 1570](#), January 1994.
- [12] Rand, D., "PPP Reliable Transmission", [RFC 1663](#), July 1994.

## Chair's Address

The working group can be contacted via the current chair:

Karl Fox  
Ascend Communications  
3518 Riverside Drive, Suite 101  
Columbus, Ohio 43221

EMail: karl@ascend.com

## Authors' Addresses

Questions about this memo should be directed to:

Kevin Schneider  
Adtran, Inc.  
901 Explorer Blvd.  
Huntsville, AL 35806-2807

Phone: (205) 971-8000  
EMail: kevin@adtran.com

Stuart Venters  
Adtran, Inc.  
901 Explorer Blvd.  
Huntsville, AL 35806-2807

Phone: (205) 971-8000  
EMail: sventers@adtran.com