Y(J) Stein
R. Shashoua
R. Insler
M. Anavi
RAD Data Communications

October 26, 2003

TDM over IP
draft-anavi-tdmoip-06.txt

Abstract

This document describes methods for transporting time division multiplexed (TDM) digital voice and data signals over Pseudowires. It is a revision of the document draft-anavi-tdmoip-05.

Table of Contents

1. Introduction

Telephony traffic is conventionally carried over connection- oriented synchronous or plesiochronous links (loosely called TDM circuits herein). With the proliferation of packet-switched networks (PSNs), integration of TDM services into a unified PSN infrastructure has become desirable. Such integration requires emulation of TDM circuits within the PSN, a function that can be carried out using Pseudo-Wires (PWs), as described in the PWE3 requirements [PWE-REQ] and architecture [PWE-ARCH] documents. This emulation must ensure QoS and voice quality similar to those of existing TDM networks as well as preserving signaling features, as described in the TDM PW requirements document [TDM-REQ].

SAToP [SAToP] is a structure-agnostic protocol for transporting TDM over PWs. SAToP completely disregards any structure that may exist in the TDM bit-stream, such as T1 or E1 framing described in [G.704], or that of the GSM Abis channel described in [TRAU]. Hence SAToP is ideal for transport of unstructured TDM data, and also eminently suitable for transport of structured TDM when there is no need to interpret or manipulate individual timeslots. In particular, SAToP is the technique of choice for PSNs with low packet loss, and for applications that do not require discrimination between timeslots nor intervention in TDM signaling.

When it is required or desirable to explicitly safeguard TDM structure, this can be accomplished in three conceptually distinct ways, namely structure-locking, structure-indication, and structure-reassembly. Structure-locking ensures that packets consist of entire TDM structures or multiples thereof. Structure-indication allows packets to contain arbitrary fragments of basic structures, and employs pointers to indicate where a structure commences. In structure-reassembly the individual timeslots are extracted and reorganized at ingress, and the original structure reassembled from the received constituents at egress.

All three methods of TDM structure preservation have their advantages. Structure-locking is described in [CESoPSN], while the present document describes TDMoIP, which specifies both structure-indication (see Section 4.1) and structure-reassembly (see Section 4.2) approaches. The necessity for two different approaches will be explained below.

Despite its name, the TDMoIP protocol herein described allows several types of PSN, including UDP over IPv4 or IPv6, MPLS, L2TPv3 over IP, or pure Ethernet. Implementation specifics for particular PSNs are discussed in Section 3. Although the protocol should be more generally called TDMoPW and its specific implementations TDMoIP,

TDMoMPLS, etc. we will use the nomenclature TDMoIP for reasons of consistency with previous versions of this draft.

2. TDMoIP Encapsulation

The overall format of TDMoIP packets is shown in the following figure.

```
+----------------+
| PSN headers    |
+----------------+
| control word   |
+----------------+
| payload        |
+----------------+
```

The PSN-specific headers contain all necessary infrastructure, and may consist of UDP/IP, L2TPv3 over IP, MPLS or layer 2 Ethernet. The PSN is assumed to be reliable enough and of sufficient bandwidth to enable transport of the required TDM data.

In addition to the aforementioned headers, an optional 12-byte RTP header may appear in order to provide a mechanism for explicit transfer of timing information in the packet. If RTP is used, the fixed RTP header described in [RTP], MUST immediately precede the control word in case of an IPv4 or IPv6 PSN, and MUST immediately follow it in the case of an MPLS PSN. The P (padding), X (header extension), CC (CSRC count), and M (marker) fields in the RTP header MUST be set to zero, and the PT values MUST be allocated from the range of dynamic values. The RTP sequence number SHOULD be identical to the sequence number in the TDMoIP control word (see below). When the TDMoIP edge devices have sufficiently accurate local clocks or can derive a sufficiently accurate timing source without explicit timestamps, the RTP header is omitted.

If a TDMoIP edge device is required to handle multiple circuit bundles, then it is the responsibility of the PSN-specific layers to provide a circuit bundle identifier (CBID) in order to enable differentiation between these circuits. A circuit bundle is defined as a stream of bits that have originated from a single physical interface or from interfaces that share a common clock, which are transmitted from a single TDMoIP source device to a single TDMoIP destination device. For example, bundles may comprise some number of 64 Kbps timeslots originating from a single E1, or an entire T3 or E3. Circuit bundles are uni-direction streams, but are universally coupled with bundles in the opposite direction to form a bi-directional connection.

The 32-bit control word MUST appear in every TDMoIP packet. Its format is given in the following figure.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FORMID |L|R|  RES  |  Length   |         Sequence Number       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

FORMID Format identifier (4 bits) is an OPTIONAL field that specifies the payload format. When it is not used it must be set to zero. The following values are presently defined:

1100 AAL1 unstructured
1101 AAL1 structured
1110 AAL1 structured with CAS
1001 AAL2
1111 HDLC

The payload format for each of these cases will be described later.

L Local Loss of Sync failure (1 bit) The L bit being set indicates that the source has detected or has been informed of a TDM physical layer fault impacting the data to be transmitted. This bit can be used to indicate Physical layer LOS that should trigger AIS generation at the far end. When the L bit is set the contents of the packet may not be meaningful, and the payload size MAY be reduced in order to conserve bandwidth. Once set, if the TDM fault is rectified the L bit MUST be cleared.

R Remote Receive failure (1 bit) The R bit being set indicates that the source is not receiving packets at its TDMoIP receive port, indicating failure of that direction of the bi-directional connection. This indication can be used to signal congestion or other network related faults. Receiving remote failure indication MAY trigger fall-back mechanisms for congestion avoidance. The R bit MUST be set after a preconfigured number of consecutive packets are not received, and MUST be cleared once packets are once again received.

RES (4 bits) These bits are reserved and MUST be set to zero.

Length (6 bits) is used to indicate the length of the TDMoIP packet (control word and payload), in case padding is employed to meet minimum transmission unit requirements of the PSN. It MUST be used if the total packet length (including PSN, optional RTP, control word, and payload) is less than 64 bytes, and MUST be set

to zero if not used.

Sequence number (16 bits) The TDMoIP sequence number provides the common PW sequencing function, and enables detection of lost packets. Since the basic clock rate for each circuit bundle is constant, the sequence number may also be used as an approximate timestamp. The initial value of the sequence number SHOULD be random (unpredictable) for security purposes, and its value is incremented modulo 2^16 separately for each circuit bundle.

3. Encapsulation Details for Specific PSNs

3.1 UDP/IP

The UDP/IP header as described in [UDP] and [IP] is prefixed to the TDMoIP data. The TDMoIP packet structure is as follows:

```
        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
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       | IPVER |  IHL  |    IP TOS     |          Total Length         |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |         Identification        |Flags|      Fragment Offset    |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |  Time to Live |    Protocol   |       IP Header Checksum      |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                       Source IP Address                       |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                    Destination IP Address                     |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       | VER |           CBID          |   Destination Port Number     |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |           UDP Length          |          UDP Checksum         |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    opt|RTV|P|X|  CC   |M|     PT      |       RTP Sequence Number     |opt
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    opt|                           Timestamp                           |opt
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    opt|                        SSRC identifier                        |opt
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |FORMID |L|R|  RES  |  Length   |          Sequence Number      |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
       |                                                               |
       |                        TDMoIP Payload                         |
       |                                                               |
       +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The first five rows are the IP header, the sixth and seventh rows are the UDP header. Rows 8 through 10 are the optional RTP header. Row 11 is the TDMoIP control word.

IPVER (4 bits) is the IP version number, e.g. for IPv4 IPVER=4.

IHL (4 bits) is the length in 32-bit words of the IP header, IHL=5.

IP TOS (8 bits) is the IP type of service.

Total Length (16 bits) is the length in octets of header and data.

Identification (16 bits) is the IP fragmentation identification field.

Flags (3 bits) are the IP control flags and MUST be set to Flags=010 to avoid fragmentation.

Fragment Offset (13 bits) indicates where in the datagram the fragment belongs and is not used for TDMoIP.

Time to Live (8 bits) is the IP time to live field. Datagrams with zero in this field are to be discarded.

Protocol (8 bits) MUST be set to 0x11 to signify UDP.

IP Header Checksum (16 bits) is a checksum for the IP header.

Source IP Address (32 bits) is the IP address of the source.

Destination IP Address (32 bits) is the IP address of the destination.

VER (3 bits) is the TDMoIP version number. The original version (VER=000) was experimental and should no longer be used. Presently VER=001 when RTP is not used, and VER=011 when RTP is used.

CBID Circuit Bundle Identifier (13 bits) This field is usually dedicated to the Source Port Number, but here identifies the unique data stream emanating from a given trunk and sharing a common destination. This nonstandard use of a UDP port number is similar to RTP/RTCP's use of port numbers to uniquely identify sessions, and the common practice (sanctioned in H.225) of randomly allocating port numbers for VoIP sessions. Here placing the circuit bundle identifier in the UDP header rather than the application area enables fast switching. The available circuit bundle numbers are 1-8063; 0 is invalid; 8191 (1FFF) is used for

OAM control messages (see Section 5); and the 127 ports 8064-8190 are reserved.

Destination Port Number (16 bits) MUST be set to 0x085E (2142), the user port number which has been assigned to TDMoIP by the Internet Assigned Numbers Authority (IANA).

UDP Length (16 bits) is the length in octets of UDP header and data.

UDP Checksum (16 bits) is the checksum of UDP/IP header and data. If not computed it must be set to zero.

3.2 MPLS

The MPLS header as described in [MPLS] is prefixed to the TDMoIP data. The packet structure (as seen at the edges) is as follows:

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Outer Label               | EXP |S| TTL           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Inner Label = CBID        | EXP |S| TTL           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FORMID |L|R|  RES  |  Length   |         Sequence Number       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                            PAYLOAD                            |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

The first two rows depicted above are the MPLS header; the third is the TDMoIP control word.

Outer Label (20 bits) is the MPLS label that identifies the MPLS LSP used to tunnel the TDM packets through the MPLS network. It is also known as the tunnel label or the transport label. The label number can be assigned either by manual provisioning or via the MPLS control protocol. While transiting the MPLS network there can be zero, one or more outer label rows. For label stack usage see [MPLS].

EXP (3 bits) experimental field

S (1 bit) stacking bit where 1 indicates stack bottom S=0 for all outer labels

TTL (8 bits) MPLS Time to live

Inner Label (20 bits) the MPLS inner label (also known as the PW label or the interworking label), contains the circuit bundle identifier used to multiplex multiple circuit bundles within the same tunnel. Valid values are as in the pervious subsection. Note that the inner label is always be at the bottom of the MPLS label stack, and hence its stacking bit is set.

3.3 L2TPv3

If L2TP is used over IPv4 without UDP the L2TPv3 header defined in [L2TPv3] is prefixed to the TDMoIP data.

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                     Session ID = CBID                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      cookie 1 (optional)                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      cookie 2 (optional)                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|FORMID |L|R|  RES  |  Length   |         Sequence Number       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                         PAYLOAD                               |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Session ID (32 bits) is the locally significant L2TP session identifier, and contains the circuit bundle identifier used to multiplex multiple circuit bundles within the same tunnel. Valid values are as in subsection 3.1 supra.

Cookie (32 or 64 bits) is an optional field that contains a randomly selected value that can be used to validate association of the received frame with the expected circuit bundle.

3.4 Ethernet

The TDMoIP packet described in the previous subsections will frequently be further encapsulated in an Ethernet frame by prefixing the Ethernet preamble, destination and source MAC addresses, optional VLAN header, etc. and appending the four octet frame check sequence after the TDMoIP frame. TDMoIP implementations MUST be able to

receive both industry standard (DIX) Ethernet and IEEE 802.3 CSMA/CD frames and SHOULD transmit Ethernet frames.

Ethernet encapsulation introduces restrictions on both minimum and maximum packet size. Whenever the entire TDMoIP packet is less than 64 bytes, zero padding is introduced and the true length indicated by using the Length field in the control word. In order to avoid fragmentation the TDMoIP packet must be restricted to the maximum payload size. For example, the length of the Ethernet payload for a non-RTP AAL2 adapted E1 trunk with 31 channels is 8*4 + 31*47 = 1489 octets. This falls below the maximal permitted payload size of 1500 bytes.

Layer 2 Ethernet frames can be directly used for TDMoIP transport without IP or MPLS layers. In this case the CBID is be carried in an MPLS-style inner label, and hence the Ethernet protocol type may be reasonably set to MPLS.

```
           +-----------------------+
           | destination address   |
           +-----------------------+
           | source address        |
           +-----------------------+
           | VLAN tag (optional)   |
           +-----------------------+
           | protocol type         |
           +-----------------------+
           | inner label           |
           +-----------------------+
           | control word          |
           +-----------------------+
           | payload               |
           +-----------------------+
           | CRC                   |
           +-----------------------+
```

4. TDMoIP Payload types

TDMoIP is a trunking application, i.e. it transports entire trunks containing multiple voice and/or data streams. Trunking can be carried out at two levels - circuit emulation and loop emulation. Circuit emulation is a structure-indication method of transporting TDM in which the TDM trunk (circuit) bit-stream is transferred across the network intact, without separation into individual timeslots. Loop emulation is a structure-reassembly method whereby the individual timeslots (loops) are identified and transported, albeit while preserving the trunk integrity.

TDMoIP uses constant-rate AAL1 [AAL1,CES] for circuit emulation, while variable-rate AAL2 [AAL2] is employed for loop emulation. Additionally, a third mode is defined specifically for transport of HDLC-based CCS signaling.

The AAL1 mode must be used for structured transport of data and is recommended for trunks with relatively constant usage. AAL2 may be used to conserve bandwidth for voice-carrying trunks in which usage is highly variable. The HDLC mode is mainly for efficient transport of trunk-associated CCS signaling.

The AAL family of protocols is a natural choice for trunking applications. Although originally developed to adapt various types of application data to the rigid format of ATM, the mechanisms are general solutions to the problem of transporting constant or variable bandwidth data streams over a packet network.

In addition, since the AAL mechanisms are extensively used within and on the edge of the telephony system, they were specifically designed for audio, non-audio data and telephony signaling.

Finally, simple service interworking with legacy networks is a major design goal of TDMoIP. Re-uses of AAL technologies simplifies interworking with existing AAL1 and AAL2 networks.

4.1 AAL1 Format Payload

For the prevalent case for which the timeslot allocation is static and no activity detection is performed, the payload can be efficiently encoded using constant bit rate AAL1 adaptation. The AAL1 format is described in [AAL1] and its use for circuit emulation over ATM in [CES]. We will herein briefly describe the use of AAL1 in the context of TDMoIP; the reader will find the full description in the normative references.

In AAL1 mode the TDMoIP payload consists of between one and thirty 48-octet subframes. The number of subframes is pre-configured and typically chosen according to latency and bandwidth constraints. Using a single subframe reduces latency to a minimum, but incurs the highest overhead, while using, for example, eight subframes reduces the overhead percentage while increasing the latency by a factor of eight.

```
+-------------+----------------+
|control word |48-octet subframe|
+-------------+----------------+
```

Single AAL1 subframe per TDMoIP packet

```
+-------------+----------------+   +-----------------+
|control word |48-octet subframe|---|48-octet subframe|
+-------------+----------------+   +-----------------+
```

Multiple AAL1 subframes per TDMoIP packet

The first octet of each 48-octet AAL1 subframe consists of an error protected three-bit sequence number.

```
 1 2 3 4 5 6 7 8
+-+-+-+-+-+-+-+-+----------------------
|C| SN  | CRC |P| 47 octets of payload
+-+-+-+-+-+-+-+-+----------------------
```

where

C (1 bit) convergence sublayer indication, its use here is limited to indication of the existence of a pointer (see below) C=0 means no pointer, C=1 means a pointer is present.

SN (3 bits) The AAL1 sequence number increments from subframe to subframe.

CRC (3 bits) is a 3 bit error cyclic redundancy code on C and SN.

P (1 bit) even byte parity

As can be readily inferred this octet can only take on eight different values, and incrementing the sequence number forms an eight subframe sequence number cycle, the importance of which will become clear shortly.

The structure of the remaining 47 octets in the TDMoIP-AAL1 subframe depends on the subframe type, of which there are three, corresponding to the three types of AAL1 circuit emulation service defined in [CES]. These are known as namely unstructured circuit emulation, structured circuit emulation and structured circuit emulation with CAS.

The simplest subframe is the unstructured one, which is used for transparent transfer of whole trunks (T1,E1,T3,E3). Although AAL1 provides no inherent advantage as compared to SAToP for unstructured

transport, in certain cases AAL1 may be required or desirable. For example, when it is necessary to interwork with an existing AAL1-based network, or when clock recovery based on AAL1-specific mechanisms is favored.

For unstructured AAL1 the 47 octets after the sequence number octet contain 376 bits from the TDM bit stream. No frame synchronization is supplied or implied, and framing is the sole responsibility of the end-user equipment. Hence the unstructured mode can be used for leased lines which carry data rather than N*64 Kbps timeslots, and even for trunks with nonstandard frame synchronization. For the T1 case the raw frame consists of 193 bits, and hence 1 183/193 T1 frames fit into each TDMoIP-AAL1 subframe. The E1 frame consists of 256 bits, and so 1 15/32 E1 frames fit into each subframe.

When the TDM trunk is segmented into timeslots according to [G704], and it is desired to transport N*64 Kbps circuit where N is only a fraction of the full E1 or T1, it is advantageous to use one of the structured AAL1 circuit emulation services. Structured AAL1 views the data not merely as a bit stream, but as a circuit bundle of timeslots. Furthermore, when CAS signaling is used it can be formatted such that it can be readily detected and manipulated.

In the structured circuit emulation mode without CAS, N octets from the N timeslots to be transported are first arranged in order of timeslot number. Thus if timeslots 2, 3, 5, 7 and 11 are to be transported the corresponding five octets are placed in the subframe immediately after the sequence number octet. This placement is repeated until all 47 octets in the subframe are taken;

```
octet    1  2  3  4  5  6  7  8  9 10 --- 41 42 43 44 45 46 47
timeslot 2  3  5  7 11  2  3  5  7 11 ---  2  3  5  7 11  2  3
```

the next subframe commences where the present subframe left off

```
octet    1  2  3  4  5  6  7  8  9 10 --- 41 42 43 44 45 46 47
timeslot 5  7 11  2  3  5  7 11  2  3 ---  5  7 11  2  3  5  7
```

and so forth. The set of timeslots 2,3,5,7,11 is called a structure and the point where one structure ends and the next commences is a structure boundary.

The problem with this arrangement is the lack of explicit indication of the octet identities. As can be seen in the above example, each TDMoIP-AAL1 subframe starts with a different timeslot, so a single lost packet will result in misidentifying timeslots from that point onwards, without possibility of recovery. The solution to this deficiency is the periodic introduction of a pointer to the next

structure boundary. This pointer need not be used too frequently, as the timeslot identification are uniquely inferable unless packets are lost.

The particular method used in AAL1 is to insert a pointer once every sequence number cycle of length eight subframes. The pointer is seven bits and protected by an even parity MSB, and so occupies a single octet. Since seven bits are sufficient to represent offsets larger than 47, we can limit the placement of the pointer octet to subframes with even sequence number. Unlike usual TDMoIP- AAL1 subframes with 47 octets available for payload, subframes which contain a pointer, called P-format subframes, have the following format.

```
     0                   1
     1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-----------------------
    |C| SN  | CRC |P|E|   pointer   | 46 octets of payload
    +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-----------------------
```

where

C (1 bit) convergence sublayer indication, C=1 for P-format subframes

SN (3 bits) is an even AAL1 sequence number

CRC (3 bits) is a 3 bit error cyclic redundancy code on C and SN

P (1 bit) even byte parity LSB for sequence number octet

E (1 bit) even byte parity MSB for pointer octet

pointer (7 bits) pointer to next structure boundary

Since P-format subframes have 46 octets of payload and the next subframe has 47 octets, viewed as a single entity the pointer needs to indicate one of 93 octets. If P=0 it is understood that the structure commences with the following octet (i.e. the first octet in the payload belongs to the lowest numbered timeslot). P=93 means that the last octet of the second subframe is the final octet of the structure, and the following subframe commences with a new structure. The special value P=127 indicates that there is no structure boundary to be indicated (needed when extremely large structures are being transported).

The P-format subframe is always placed at the first possible position in the sequence number cycle that a structure boundary occurs, and

can only occur once per cycle.

The only difference between the structured circuit emulation format and structured circuit emulation with CAS is the definition of the structure. Whereas in structured circuit emulation the structure is composed of the N timeslots, in structured circuit emulation with CAS the structure encompasses the superframe consisting of multiple repetitions of the N timeslots and then the CAS signaling bits. The CAS bits are tightly packed into octets and the final octet is padded with zeros if required.

For example, for E1 trunks the CAS signaling bits are updated once per superframe of 16 frames. Hence the structure for N*64 derived from an E1 with CAS signaling consists of 16 repetitions of N octets, followed by N sets of the four ABCD bits, and finally four zero bits if N is odd. For example, the structure for timeslots 2,3 and 5 will be as follows

```
2 3 5 2 3 5 2 3 5 2 3 5 2 3 5 2 3 5 2 3 5 2 3 5 2 3 5 2 3 5 2 3 5
2 3 5 2 3 5 2 3 5 2 3 5 2 3 5 [ABCD2 ABCD3] [ABCD5 0000]
```

Similarly for T1 ESF trunks the superframe is 24 frames, and the structure consists of 24 repetitions of N octets, followed by the ABCD bits as before. For the T1 case the signaling bits will in general appear twice, in their regular (bit-robbed) positions and at the end of the structure.

4.2 AAL2 Format Payload

Although AAL1 may be configured to transport fractional trunks, the allocation of timeslots to be transported must be static due to the fact that AAL1 is a constant rate bit-stream. It is often the case that not all the timeslots in a trunk are simultaneously active ("off-hook"), and by observation of the TDM signaling timeslot activity status may be determined. Moreover, even during active calls there is silence about half the time. Using the variable rate AAL2 mode we may dynamically allocate timeslots to be transported, thus conserving bandwidth.

The variable rate AAL2 format is described in [AAL2] and its use for loop emulation over ATM is explained in [SSCS,LES].

For TDMoIP the AAL2 streams need not be segmented into ATM cells, rather the AAL2 payloads belonging to all timeslots are concatenated, and a single packet sent over the network.

```
        +-------------+------------+   +-------------+
        |control word |AAL2 subframe|---|AAL2 subframe|
        +-------------+------------+   +-------------+
```

Concatenation of AAL2 subframes in a TDMoIP packet

The basic AAL2 subframe is :

```
      |    Octet 1    |    Octet 2    |    Octet 3    |
       0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7 0 1 2 3 4 5 6 7
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+------------
      |      CID      |     LI    |   UUI   |   HEC   |   PAYLOAD
      +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+------------
```

CID (8 bits) channel identifier is a unique identifier for the bundle. The values below 8 are reserved and so there are 248 possible channels. The mapping of CID values to trunk timeslots is outside the scope of the TDMoIP protocol and must be configured manually or via network management.

LI (6 bits) length indicator is one less than the length of the payload in octets. (Note that the payload is limited to 64 octets.)

UUI (5 bits) user-to-user indication is the higher layer (application) identifier and counter. For voice data the UUI will always be in the range 0-15, and SHOULD be incremented modulo 16 each time a channel buffer is sent. The receiver MAY monitor this sequence. UUI is set to 24 for CAS signaling packets.

HEC (5 bits) the header error control

Payload - voice A block of length indicated by LI of voice samples are placed as- is into the AAL2 packet.

Payload - CAS signaling For CAS signaling the payload is formatted as a type 3 packet (in the notation of [AAL2]) in order to ensure error protection. The signaling is sent with the same CID as the corresponding voice channel. Signaling is sent whenever the state of the ABCD bits changes, and is sent with triple redundancy, i.e. sent three times spaced 5 milliseconds apart. In addition, the entire set of the signaling bits is sent periodically to ensure reliability.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|RED|       timestamp             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  RES  | ABCD  |    type   | CRC
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
    CRC (cont)  |      PAD        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

RED (2 bits) is the triple redundancy counter. For the first packet it takes the value 00, for the second 01 and for the third 10. RED=11 means non-redundant information and is used for periodic refresh of the CAS information.

Timestamp (14 bits) The timestamp is the same for all three redundant transmissions.

RES (4 bits) is reserved and MUST be set to zero

ABCD (4 bits) are the CAS signaling bits

type (6 bits) for CAS signaling this is 000011

CRC-10 (10 bits) is a 10 bit CRC error detection code

PAD (8 bits) is set to zero.

[PWE-ARCH] denotes as Native Service Processing (NSP) functions all processing of the TDM data before its use as payload. Since AAL2 is inherently variable rate, arbitrary NSP functions MAY be performed before the timeslot is placed in the AAL2 loop emulation payload. This includes testing for on-hook/off-hook status, voice activity detection, speech compression, fax/modem relay, etc.

4.3 HDLC Format Payload

The motivation for handling HDLC in TDMoIP is to efficiently transport CCS (common channel signaling such as SS7) which is embedded in the TDM stream. This mechanism is not intended for general HDLC payloads, and assumes that the HDLC messages are always shorter than the maximum packet size.

The HDLC format is intended to operate in port mode, transparently passing all HDLC data and control messages over the PW.

In order to transport HDLC the sender monitors flags until a frame is detected. The contents of the frame are collected and the FCS tested. If the FCS is incorrect the frame is discarded, otherwise the frame is sent after initial or final flags and FCS have been

discarded and bit unstuffing has been performed. When an TDMoIP-HDLC frame is received its FCS is calculated, and the original HDLC frame reconstituted.

5. OAM Signaling

Since the TDMoIP PW is not absolutely reliable, it requires a signaling mechanism to provide feedback regarding problems in the communications environment. In addition, such signaling can be used to collect statistics relating to the performance of the underlying PSN [IPPM].

If the underlying PSN has adequate signaling mechanisms then these are to be used. If not, the ICMP-like procedures detailed below SHOULD be followed.

All TDMoIP OAM signaling messages MUST use CBID 8191 (1FFF). All PSN layer parameters (for example, IP addresses, TOS, EXP bits, and VLAN ID) MUST remain those of the circuit bundle being investigated.

5.1 Connectivity-Check Messages

In most conventional IP applications a server sends some finite amount of information over the network after explicit request from a client. With TDMoIP the source sends a continuous stream of packets towards the destination without knowing whether the destination device is ready to accept them, leading to flooding of the PSN.

The problem may occur when an edge device fails or is disconnected from the PSN, or the PW is broken. After an aging time the destination edge disappears from the routing tables, and intermediate routers may flood the network with the TDMoIP packets in an attempt to find a new path.

The solution to this problem is to significantly reduce the number of TDMoIP packets transmitted per second when PW failure is detected, and to return to full rate only when the PW is restored. The detection of failure and restoration is made possible by the periodic exchange of one-way connectivity-check messages, as defined in [CONNECT].

Connectivity is tested by periodically sending OAM messages from the source edge to the destination edge, and having the destination reply to each message. The format of connectivity- check messages is given in subsection 10.3 infra.

The connectivity check mechanism can also be useful during setup and configuration. Without OAM signaling one must ensure that the

destination edge is ready to receive packets before starting to send them. Since TDMoIP edge devices usually operate full-duplex, both edges must be set up and properly configured simultaneously if flooding is to be avoided. By using the connectivity mechanism a configured edge device waits until it can detect its destination before transmitting at full rate. In addition, errors in configuration can be readily discovered by using the service specific field.

5.2 Performance Measurements

In addition to one way connectivity, the OAM signaling mechanism can be used to request and report on various PSN metrics, such as one way delay, round trip delay, packet delay variation, etc. It can also be used for remote diagnostics, and for unsolicited reporting of potential problems (e.g. dying gasp messages).

5.3 OAM Packet Format

The format of an OAM message packet is depicted in the following figure. Note that PSN-specific layers are identical to those used to carry the TDMoIP data, with the exception that their CBID = 1FFF instead of the usual circuit bundle identifier.

```
 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
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |         PSN-specific layers  (with CBID=1FFF)                 |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |FORMID |L|R|  RES  | Length    |      OAM Sequence Number      |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 | OAM Msg Type  | OAM Msg Code  | Service specific information  |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |          Source CBID          |        Destination CBID       |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                   Source Transmit Timestamp                   |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                 Destination Receive Timestamp                 |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
 |                Destination Transmit Timestamp                 |
 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

FORMID, L and R are identical to those used for the circuit bundle being tested.

Length is the length in bytes of the OAM message packet.

OAM Sequence Number (16 bits) is used to uniquely identify the

message. Its value is unrelated to the sequence number of the TDMoIP data packets for the circuit bundle in question. It is incremented in query messages, and replicated without change in replies.

OAM Msg Type (8 bits) indicates the function of the message. At present the following are defined:

0 for one way connectivity query message
8 for one way connectivity reply message.

OAM Msg Code (8 bits) is used to carry information related to the message, and its interpretation depends on the message type. For type 0 (connectivity query) messages the following codes are defined:

0 validate connection.
1 do not validate connection

for type 8 (connectivity reply) messages the available codes are:

0 acknowledge valid query
1 invalid query (configuration mismatch).

Service specific information (16 bits) is a field that can be used to exchange configuration information between edge devices. If it is not used this field MUST contain zero. Its interpretation depends on the FORMID field. At present the following is defined for AAL1 payloads.

```
 0                   1
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Number of TSs | Number of SFs |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Number of TSs (8 bits) is the number of timeslots being transported, e.g. 24 for full T1.

Number of SFs (8 bits) is the number of 48-octet AAL1 subframes per packet, e.g. 8 when packing 8 subframes per packet.

Source CBID (16 bits) uniquely identifies the circuit bundle as labeled by the source edge.

Destination CBID (16 bits) uniquely identifies the circuit bundle as labeled by the destination edge.

Source Transmit Timestamp (32 bits) represents the time the source edge transmitted the query message in units of 100 microseconds. This field and the following ones only appear if delay is being measured.

Destination Receive Timestamp 32 bits) represents the time the destination edge received the query message in units of 100 microseconds.

Destination Transmit Timestamp (32 bits) represents the time the destination edge transmitted the reply message in units of 100 microseconds.

6. Implementation Issues

General requirements for transport of TDM over pseudo-wires are detailed in [TDM-REQ]. In the following subsections we review additional aspects essential to successful TDMoIP implementation.

6.1 Quality of Service

TDMoIP does not provide mechanisms to ensure timely delivery or provide other quality-of-service guarantees; hence it is required that the lower-layer services do so. Layer 2 priority can be bestowed upon a TDMoIP stream by using the VLAN priority field, MPLS priority can be provided by using EXP bits, and layer 3 priority is controllable by using TOS. Switches and routers which the TDMoIP stream must traverse should be configured to respect these priorities.

If the PSN is Diffserv-enabled then an EF-PHB (expedited forwarding) class based PDB SHOULD be used, in order to provide a low latency and minimal jitter service. It is suggested that the transport LSP be somewhat overprovisioned.

If the MPLS network is Intserv enabled, then GS (Guaranteed Service) with the appropriate bandwidth reservation SHOULD be used in order to provide a bandwidth BW guarantee equal or greater than that of the aggregate TDM traffic. The delay introduced by the MPLS network SHOULD be measured prior to traffic flow, to ensure its compliance with latency requirements.

6.2 Timing

TDM networks are inherently synchronous; somewhere in the network there will always be at least one extremely accurate primary reference clock, with long-term accuracy of one part in 10E-11. This

node, whose accuracy is called "stratum 1", provides reference timing to secondary nodes with lower "stratum 2" accuracy, and these in turn provide reference clock to "stratum 3" nodes. This hierarchy of time synchronization is essential for the proper functioning of the network as a whole; for details see [G823,G824]. The use of time standards less accurate than stratum 4 is NOT RECOMMENDED as it may result in service impairments.

Packets in IP networks reach their destination with delay that has a random component, known as jitter. When emulating TDM on a PSN, it is possible to overcome this randomness by using a "jitter buffer" on all incoming data, assuming the proper time reference is available. The problem is that the original TDM time reference information is not disseminated through the PSN.

In broadest terms there are two methods of overcoming this difficulty; in one the timing information is provided by some means independent of the PSN, while in the other the timing must be transferred over the PSN.

For example, if the entire TDM infrastructure (or at least major portions of it) is replaced by TDMoIP timing information MUST be delivered over the PSN, and the reconstructed TDM stream MUST still conform to ITU-T recommendations [G823] for E1 and [G824] for T1 trunks.

However, TDMoIP is frequently used in a "toll-bypass" scenario, where a PSN link connects two existing TDM networks. In such a case both TDMoIP devices MUST receive accurate timing from the TDM networks to which they connect, and MUST use this local timing when outputting to the TDM network.

6.3 Jitter and Packet Loss

In order to compensate for packet delay variation that exists in any IP network a jitter buffer MUST be provided. The length of this buffer SHOULD be configurable and MAY be dynamic (i.e. grow and shrink in length according to the statistics of the delay variation).

In order to handle (infrequent) packet loss and misordering a packet order integrity mechanism MUST be provided. This mechanism MUST track the serial numbers of packets in the jitter buffer and MUST take appropriate action when faults are detected. When missing packet(s) are detected the mechanism MUST output interpolation packet(s) in order to retain TDM timing. Packets with incorrect serial numbers or other detectable header errors MAY be discarded. Packets arriving in incorrect order SHOULD be swapped. Whenever possible, interpolation packets SHOULD ensure that proper

synchronization bits are sent to the TDM network.

While the insertion of arbitrary interpolation packets may be sufficient to maintain the TDM timing, for voice traffic packet loss can cause in gaps or artifacts that result in choppy, annoying or even unintelligible speech, see [TDM-PLC]. An implementation MAY blindly insert a preconfigured constant value in place of any lost speech samples, and this value SHOULD be chosen to minimize the perceptual effect. Alternatively one MAY replay the previously received packet. Since a TDMoIP packet is usually declared lost following the reception of the next packet, when computational resources are available, implementations SHOULD conceal the packet loss event by estimating the missing sample values.

7. Security Considerations

TDMoIP does not enhance or detract from the security performance of the underlying PSN, rather it relies upon the PSN's mechanisms for encryption, integrity, and authentication whenever required.

TDMoIP does not provide protection against malicious users utilizing snooping or packet injection during setup or operation. However, random initialization of sequence numbers makes known-plaintext attacks on link encryption methods more difficult.

Circuit bundle identifiers SHOULD be selected in an unpredictable manner rather than sequentially or otherwise in order to deter session hijacking. When using L2TP randomly selected cookies MAY be used to validate circuit bundle origin. Sequence numbers SHOULD be randomly initialized in order to increase the difficulty of decrypting based on packet headers.

8. IANA Considerations

When used with UDP/IP the destination port number MUST be set to 0x085E (2142), the user port number which has been assigned by the to TDMoIP.

The format identifiers (FORMID) will need to be standardized.

9. Normative References

[AAL1] ITU-T Recommendation I.363.1 (08/96) B-ISDN ATM Adaptation Layer (AAL) specification: Type 1

[AAL2] ITU-T Recommendation I.363.2 (11/00) B-ISDN ATM Adaptation Layer (AAL) specification: Type 2

[CES] ATM forum specification atm-vtoa-0078 (CES 2.0) Circuit Emulation Service Interoperability Specification Ver. 2.0

[CONNECT] RFC 2678 IPPM Metrics for Measuring Connectivity

[DELAY] RFC 2679 A One-way Delay Metric for IPPM

[G704] ITU-T Recommendation G.704 (10/98) Synchronous frame structures used at 1544, 6312, 2048, 8448 and 44736 Kbit/s hierarchical levels

[G823] ITU-T Recommendation G.823 (03/00) The control of jitter and wander within digital networks which are based on the 2048 Kbit/s hierarchy

[G824] ITU-T Recommendation G.824 (03/00) The control of jitter and wander within digital networks which are based on the 1544 Kbit/s hierarchy

[IPPM] RFC 2330 Framework for IP Performance Metrics

[IPv4] RFC 791 (STD0005) Internet Protocol (IP)

[LES] ATM forum specification atm-vmoa-0145 (LES) Voice and Multimedia over ATM - Loop Emulation Service Using AAL2

[L2TPv3] draft-ietf-l2tpext-l2tp-base-10.txt (08/03) Layer Two Tunneling Protocol (L2TPv3), J. Lau et al., work in progress

[MPLS] RFC 3032 MPLS Label Stack encoding

[RTP] RFC 3550 RTP: Transport Protocol for Real-Time Applications

[SAToP] draft-ietf-pwe3-satop-00.txt (09/03) Structure-Agnostic TDM over Packet (SAToP), A. Vainshtein and Y. Stein, work in progress

[SSCS] ITU-T Recommendation I.366.2 (02/99) AAL Type 2 service specific convergence sublayer for trunking

[TRAU] GSM 08.60 (10/01) Digital cellular telecommunications system (Phase 2+); Inband control of remote transcoders and rate adaptors for Enhanced Full Rate (EFR) and full rate traffic channels

[UDP] RFC 768 (STD0006) User Datagram Protocol (UDP)

10. Informative References

[CESoPSN] draft-vainshtein-cesopsn-06.txt (03/03), TDM Circuit Emulation Service over Packet Switched Network, A. Vainshtein et al, work in progress

[PWE3-ARCH] draft-ietf pwe3-arch-06.txt (10/03), PWE3 Architecture, Stewart Bryant et al, work in progress

[PWE3-REQ] draft-ietf-pwe3-requirements-06.txt (12/03) Requirements for Pseudo Wire Emulation Edge-to-Edge (PWE3), XiPeng Xiao et al, work in progress

[TDM-PLC] draft-stein-pwe3-tdm-packetloss-01.txt (10/03), The Effect of Packet Loss on Voice Quality for TDM over Pseudowires, Y(J) Stein and I. Druker, work in progress

[TDM-REQ] draft-ietf-pwe3-tdm-requirements-01.txt (12/03), Requirements for Edge-to-Edge Emulation of TDM Circuits over Packet Switching Networks, M. Riegel et al., work in progress

11. Acknowledgments

The authors would like to thank Hugo Silberman, Shimon HaLevy, Tuvia Segal, and Eitan Schwartz of RAD Data Communications for their valuable contributions to the technology described herein.

Authors' Addresses

Yaakov (Jonathan) Stein
RAD Data Communications
24 Raoul Wallenberg St., Bldg C
Tel Aviv 69719
ISRAEL

Phone: +972 3 645-5389
EMail: yaakov_s@rad.com

Ronen Shashoua
RAD Data Communications
24 Raoul Wallenberg St., Bldg C
Tel Aviv 69719
ISRAEL

Phone: +972 3 645-5447
EMail: ronen_s@rad.com

Ron Insler
RAD Data Communications
24 Raoul Wallenberg St., Bldg C
Tel Aviv 69719
ISRAEL

Phone: +972 3 645-5445
EMail: ron_i@rad.com

Motty (Mordechai) Anavi
RAD Data Communications
900 Corporate Drive
Mahwah, NJ 07430
USA

Phone: +1 201 529-1100 Ext. 213
EMail: motty@radusa.com

Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.