
Leuven – 14-18 June 2004

Question: 4/15

SOURCE¹: Texas Instruments, Inc.TITLE: VDSL2 – **Constraining the Interleaver Complexity**

Abstract

This contribution proposes restrictions on the interleaver. The interleaver is a major source of complexity in VDSL2. We propose that the interleaver delay in time be restricted rather than restricting the depth as in ADSL2. This allows the following: 1) the flexibility of using shorter codewords to correct longer bursts, 2) the capability to correct repetitive impulse noise, and 3) lower complexity implementations for profiles that do not require the full VDSL2 data rate. We propose also that the upper limit on the number of codewords in a DMT symbol (or per unit time) scale with the data rate so that more codewords are allowed at higher data rates.

[Introduction](#), [Limits on Interleaver Complexity](#), [Limits on the Number of Codewords](#), [Repetitive Impulse Noise](#), [Examples](#), [Proposal](#), [References](#)

Differences

As a courtesy to those who may have read T1E1.4/2003-493, this section lists the [differences](#) between this contribution and that one.

- generally, this contribution proposes that [the number of codewords in a given amount of time](#) and the interleaver complexity should both scale with the data rate
- the substance of the introduction has not changed
- section 2 (limits on interleaver complexity): [an example is included to illustrate practically how this proposal would work in the recommendation](#)
- section 3 (limits on the number of codewords): [this section has been completely revised](#)
- section 4 (repetitive impulse noise): this section has not changed
- section 5 (examples) [the table has been revised](#)
- section 6 (proposal) [the proposals have been revised](#)
- section 7 (references) [the references have been updated](#)

1. Introduction

A convolutional interleaver was first described by Ramsey [1] and Forney [2]. In general, a convolutional interleaver imposes a different delay on each input symbol (normally an octet). If i denotes the octet index within a

¹ Contact: Cory Modlin
Texas Instruments, Inc.

T: +1 301 318 2679

E: cmodlin@ti.com

group of I octets so that $i = 0, 1, \dots, I-1$, octet i experiences a delay of $i \cdot (d-1)$ with d the interleaver depth. The deinterleaver performs the inverse operation delaying octet i by $(I-i-1) \cdot (d-1)$. The overall delay of the interleaver/deinterleaver pair is

$$\text{interleaver delay (octets)} = (I-1) \cdot (d-1) \text{ octets.} \quad (1)$$

This applies to all interleavers being considered for VDSL2.

The smallest amount of memory required to build an interleaver/deinterleaver pair is equal to the total delay of the interleaver/deinterleaver [3]. Typically, for memory optimized interleavers, the interleaver and deinterleaver memory size is nearly the same. Therefore, the smallest possible memory for either the interleaver or deinterleaver is smallest possible (de)interleaver memory = $(I-1) \cdot (d-1)/2$ bytes.

$$(2)$$

In a typical implementation, slightly more memory is often required. The actual amount of required memory is implementation specific.

The length of a burst that can be corrected by the combination of Reed-Solomon coding and interleaving/deinterleaving is dependent on the line data rate. We define ldr as the line data rate, the rate of the Reed-Solomon encoded bits. This is as opposed to the net data rate, ndr , the effective payload data rate as seen at the $\alpha(\beta)$ interface between the TPS-TC and the PMS-TC. We adopt the notation ldr_mbit_s or ldr_kbit_s to indicate whether the data rate is given in mbit/s or kbit/s respectively. The line data rate and net data rate are related by the equation

$$ndr_kbit_s = ldr_kbit_s \cdot (n-r)/n - \text{overhead rate} \quad (3)$$

where the Reed-Solomon codeword size is n octets with r octets of redundancy.

For a t error correcting (typically $t = r/2$) Reed-Solomon code of size n octets, and assuming $n = I \cdot q^2$ the combination of coding and interleaving can correct a burst of

$$INP_min = t \cdot d/q \text{ octets} \Rightarrow t \cdot d/q \cdot 8/ldr_mbit_s \mu s. \quad (4)$$

In ADSL2 and ADSL2+, $q = 1$, t is up to 8 (assuming erasure decoding is not used), n is up to 255 and d can be up to 64 downstream and 8 upstream. In ADSL2+, the maximum line data rate is 24.48 Mbps. At this rate the coding plus interleaving can correct up to $8 \cdot 64/1 \cdot 8/24.48 = 167 \mu s$.

The end-to-end delay of the interleaver/deinterleaver in ms is

$$\text{delay (ms)} = (I-1) \cdot (d-1) \cdot 8/ldr_kbit_s \quad (5)$$

and for ADSL2+, the delay at the maximum line data rate is $(255-1) \cdot (64-1) \cdot 8/24480 = 5.23$ ms.

Error bursts must be separated in time so that each codeword corrects only one burst. The span of a codeword of size n octets and an interleaver depth, d , is

$$\text{span (ms)} = n \cdot d/q \cdot 8/ldr_kbit_s \quad (6)$$

The time spacing between codewords in equation (6) is nearly identical to the interleaver delay in equation (5) for large codeword size and depth since $n = I \cdot q$.

At the maximum codeword size, interleaver depth, and line data rate in ADSL2+, the codeword spans $(255) \cdot (64) \cdot 8/24480 = 5.33$ ms. At the maximum line data rate and codeword size, the ADSL2+ coding + interleaver can correct a $167 \mu s$ burst every 5.33 ms.

Equations (2), (4), (5), and (6) illustrate trade-offs between interleaver memory, error correction capability, delay, and burst separation. More interleaver memory normally allows more error correction but leads to higher delays and

² Here, “ q ” is assumed to be an integer. This is not the same q used to describe the DMT tone spacing.

a longer separation between error bursts. Significant error correction can be achieved by using shorter codewords requiring less memory, less delay, and shorter time between bursts. However, small codewords typically have lower net coding gain and higher computation requirements since there are more decoder operations required in the same amount of time. Therefore, we make a trade-off between complexity, capability, and performance.

2. Limits on Interleaver Complexity

The size of the interleaver memory will be a major source of complexity in VDSL2. 100 Mbit/s symmetric has been an often stated goal for VDSL2. However, a number of operators have stated their requirements at well below 100 Mbit/s. See for example [4] or [5].

Therefore, it seems prudent to define the interleaver complexity requirements in a way that will allow those who want to deploy VDSL2 at lower speeds to do so at a reduced complexity with respect to higher speed implementations.

The way to do this and to guarantee some minimum level of performance is to specify the interleaver complexity in terms of the delay in time. From equation (5), we see that the delay in time (ms) is proportional to the interleaver depth and to the codeword size and inversely proportional to the data rate.

ADSL2 instead specifies the smallest maximum interleaver depth and the maximum number of codewords allowed in a DMT symbol. There are two problems with this approach. The first is that it removes the flexibility of trading codeword size and interleaver depth to allow more error correction with the same amount of memory. The second problem is that as the data rate increases, the interleaver delay decreases and with it, the error correction capability decreases also.

One possible way to specify the smallest maximum supported delay is to start with ADSL2+ and require that VDSL2 interleavers support *at least* 5.23ms delay. This maintains a level of impulse noise protection as the data rate increases and still allows lower speed implementations to save complexity.

For interoperability reasons, the VTU-O and VTU-R must exchange the interleaver delay in terms of octets. The requirement is that the interleaver delay in octets be sufficient to satisfy the smallest maximum delay even at the highest supported data rate. If a VDSL2 implementation supports a larger interleaver memory than is required, it should be free to specify the larger value. The VTU-O and VTU-R would then select the smaller of the transmitter and receiver capabilities, in each direction, as the end-to-end capabilities.

Again, the actual amount of memory required is implementation specific.

Example:

Suppose a VDSL2 transceiver supports up to 44.5 Mbit/s as a line data rate. If the minimum interleaver delay requirement were 5.23ms, then, from equation (5) and equation (1), this transceiver must support a delay of at least 29092 octets which corresponds to having an interleaver memory of *at least* 14546 octets according to equation (2) (although this is actually implementation specific). During initialization, the VDSL2 transceiver would indicate that it could support up to 44.5 Mbit/s and 29092 or more octets of interleaver delay. The actual interleaver delay could be considerably higher than 5.23ms, in this example, if the actual connection data rate is below 44.5 Mbit/s. For example, if the actual connection line rate were 5 Mbit/s, the interleaver delay could be as high as 47 ms using 14546 octets of interleaver memory. This is why the maximum delay is still needed.

Transceiver capabilities exchanged during initialization for this example:

- **maximum data rate = 44.5 Mbit/s**
- **minimum data rate = any value at or below the maximum data rate**
- **maximum delay supported $\geq$ 29092 octets**
 - meets 5.23 ms example minimum requirement
 - minimum amount of memory required for (de)interleaver is 14546 octets
- **maximum delay = any valid value as in ADSL2 today §**

3. Limits on the Number of Codewords

Similar to the interleaver delay, we propose that the maximum number of codewords in a DMT symbol (or per unit time) also scale with the data rate. The higher the data rate, the more DMT codewords there can be in a fixed length DMT symbol.

In ADSL2+, 5.23ms delay allows correction of an impulse burst of only 167 μ s. This is because the (de)interleaver depth is limited to 64. By removing the depth restriction, the correction capability can be increased without increasing the delay in ms or the size of the (de)interleaver.

The correction capability can be enhanced by using smaller codewords. This implies there are more codewords in each DMT symbol and a larger interleaver depth. Typically the complexity increase from adding interleaver memory is considerably higher than the complexity from decoding small codewords.

Small codewords with the codeword size, n , less than 255 can lead to slightly lower net coding gain depending on the situation. As the codeword size decreases, the gross coding gain increases since a higher percentage of errors are corrected. However, at the same time, there is more overhead. Typically for very small codeword sizes, the penalty from the overhead starts to outweigh the added benefits from more error correction capability and we see a net coding gain loss. A trade-off needs to be made between performance, delay, and complexity. This trade-off needs to take into account that a majority of lines are not afflicted by large impulse noise.

At a given interleaver delay expressed in time and a given amount of impulse noise protection, the maximum codeword size allowed is fixed. To see this, we re-write equation (4) as

$$l_{dr_mbit_s} = \frac{t \cdot \frac{d}{q} \cdot 8}{INP_min_ \mu s} \quad (7)$$

and then insert this into equation (5) and find

$$\begin{aligned}
\text{delay}(ms) &= \frac{\left(\frac{n}{q} - 1\right)(d-1) \cdot 8 \cdot \text{INP_min_ms}}{t \cdot \frac{d}{q} \cdot 8} \\
&\cong \frac{\frac{n}{q} \cdot d \cdot 8 \cdot \text{INP_min_ms}}{t \cdot \frac{d}{q} \cdot 8} \\
&= \frac{n \cdot \text{INP_min_ms}}{t}
\end{aligned} \tag{8}$$

where in the second line we assume that n/q and $d \gg 1$ and we have substituted $I = n/q$. As we see, at a fixed error correction capability, t , and a fixed impulse noise correction requirement, INP_min_ms , the codeword size n is constrained by the delay.

With a fixed codeword size, the number of codewords in a fixed time period or in a DMT symbol will scale with the data rate. Therefore, we propose that the constraint on the number of codewords per unit time (or for a DMT symbol) scale with data rate.

For example, to achieve $\text{INP_min_}\mu\text{s} = 500$ (2 DMT symbols if the DMT symbol rate is 4000) at $t = 8$ and a delay constraint of 5.23ms, we need $n \approx 85$. The number of codewords in a 4 kHz DMT symbol is

$$\begin{aligned}
\text{codewords per 4 kHz DMTsymbol} &= \text{ldr_kbit_s} \cdot \frac{1 \text{ octet}}{8 \text{ bits}} \cdot \frac{1 \text{ codeword}}{n \text{ octets}} \cdot \frac{1 \text{ s}}{4 \text{ k DMT symbols}} \\
&= \frac{\text{ldr_kbit_s}}{32 \cdot n} \\
&= \frac{\text{ldr_kbit_s}}{2720}
\end{aligned} \tag{9}$$

At a line data rate of 30 mbit/s (30000 kbit/s), for example, the number of codewords per 4 kHz DMT symbol is $\frac{30000}{2720} = 11.03 \longrightarrow 12$.

Putting this in ADSL2 terms with S defined as the number of DMT symbols per codeword, we would say

$$S \geq \frac{2720}{\text{ldr_kbit_s}} \tag{10}$$

where $n = 85$ and $t = 8$ was used to constrain S . For the example at 30 Mbit/s, this would be $S \geq 1/12$.

Since variable tone spacing and a variable cyclic extension length will cause the DMT symbol period to vary from 4000 Hz, this rule on the number of codewords per DMT symbol needs to be specified in a way that allows a variable DMT symbol period. We leave this for a future meeting.

This does not mean that $n = 85$ would be the largest or smallest codeword size allowed, it is simply used to guarantee a certain minimum level of impulse noise protection at a given minimum interleaver delay at a given maximum data rate. If the actual interleaver delay is higher or the data rate lower or the impulse noise protection lower, larger codewords can be used. The proposal is that $S_{\min}$ scale with the data rate. In ADSL2, $S_{\min}$ is fixed at $1/2$ and in ADSL2+, $S_{\min}$ is fixed at $1/3$.

4. Repetitive Impulse Noise

Recently, impulse noise measurements taken by BT [6] have shown that some faulty or poorly designed consumer electronic equipment can emit 100 Hz noise pulses. Presumably, in North America, we would find similar 120 Hz noise sources although we are not aware of a comparable study. If 120 Hz noise sources exist, the period of the disturbance would be 8.33ms. To correct an impulse that occurs every 8.33ms, each Reed-Solomon codeword must span no more than 8.33ms or be designed so that each codeword can correct multiple bursts.

Noting the similarity between equations (5) and (6), as long as the length of each impulse falls within the correction capability of the Reed-Solomon code plus interleaver, the impulse train can be corrected as long as the interleaver delay is less than about 8.33ms in regions with 60Hz power transmission or 10ms in regions that use 50Hz power transmission.

5. Examples

net data rate (<i>ndr_mbit_s</i>)	line data rate (<i>ldr_mbit_s</i>)	delay(ms)	interleaver depth (<i>d</i>) for $q = 1$	number of codewords per 4000Hz DMT symbol ($1/S$)	optimal (de)interleaver memory size (octets) $[(I-1)*(d-1)/2]$	impulse noise protection
100	123.5	5.18	966	46	40048	500 μ s
36	44.5	5.17	348	17	14400	500 μ s
25	30.1	5.17	242	12	10002	500 μ s
20	24.4	5.23	191	9	8001*	500 μ s
* ADSL2/2+ minimum memory requirement.						

6. Proposal

This contribution addresses

11.4	Open	What shall be the mandatory interleaver capabilities?	MC-086, D1060
------	------	---	---------------

and proposes:

- interleaver complexity should be specified in terms of a time delay (ms), not in terms of an amount of memory or an interleaver depth
- interleaver delay in terms of octets should be exchanged between the VTU-O and VTU-R; the delay in octets should meet the minimum requirements in terms of the delay in time
- the upper limit on the number of codewords per unit time should be constrained and should scale with the data rate so that as the data rates increase, this upper limit on the number of codewords is also higher

7. References

- [1] Ramsey, J.L., "Realization of Optimum Interleavers", in *IEEE Trans Info Theory*, Vol IT-16, No. 3, May 1970, pp. 338-345.
- [2] Forney, G.D., "Burst-Correcting Codes for the Classic Bursty Channel", *IEEE Trans Communications Technology*, Vol COM-19, October 1971, pp. 772-781.
- [3] Heegard, C. and Wicker, S.B., "Turbo Coding," Kluwer Academic Publishers, Boston, MA, 1999.

- [4] Foster, K., "VDSL2 Requirements," ITU Temporary Document SS-058, January 2004.
- [5] Starr, T. and Wei, D., "VDSL2 Rate and Reach Goals," ITU Delayed Contribution 1045, April 2004.
- [6] Foster, K., "Improved Impulse Noise Modelling for xDSL Modem Testing," ITU Delayed Contribution 1251, April 2004.