

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CommScope, Inc

Petitioner

v.

TQ Delta LLC,

Patent Owner

Case: IPR2023-00064

U.S. Patent No. 8,276,048

DECLARATION OF RICHARD WESEL

TABLE OF CONTENTS

II.	Qualifications.....	2
III.	Scope of the Engagement.....	4
IV.	Legal Standards Relied Upon	6
V.	Technical Background.....	10
	a. Reed-Solomon Coding	11
	b. Interleaving.....	12
	c. Shared Memory.....	16
	d. Configuration Messages	16
VI.	The '048 Patent	18
VII.	Prosecution History	20
VIII.	Prior Decisions by the District Court.....	22
IX.	Litigation Related to the '048 Patent	23
X.	Claim Construction.....	23
	a. “transceiver”	24
	b. “shared memory”	24
	c. “amount of memory”	25
	d. “the shared memory allocated to the [deinterleaver / interleaver] is used at the same time as the shared memory allocated to the [interleaver/ deinterleaver]”	25
XI.	Prior Art References.....	25
	e. Mazzoni Overview	25
	f. Fadavi-Ardekani Overview.....	28
	g. VDSL1 Overview	30
	h. Motivation to Combine the Prior Art	33
	i. Motivation to Combine Mazzoni and VDSL1	33
	ii. Motivation to Combine VDSL1 and Fadavi-Ardekani	38
XII.	Ground 1: The combination of Mazzoni and VDSL1 renders the challenged claims obvious.....	42
	a. Analysis	42
	i. Claim 1: [1.pre] A system that allocates shared memory comprising:	42
	ii. Claim 1: [1.A] a transceiver that is capable of: transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver;.....	51

iii.	Claim 1: [1.B] determining an amount of memory required by the interleaver to interleave a first plurality of Reed Solomon (RS) coded data bytes within the shared memory;	61
iv.	Claim 1: [1.C] allocating a first number of bytes of the shared memory to the interleaver to interleave the first plurality of Reed Solomon (RS) coded data bytes for transmission at a first data rate, wherein the allocated memory for the interleaver does not exceed the maximum number of bytes specified in the message;	79
v.	Claim 1: [1.D] allocating a second number of bytes of the shared memory to a deinterleaver to deinterleave a second plurality of RS coded data bytes received at a second data rate; and	88
vi.	Claim 1: [1.E] interleaving the first plurality of RS coded data bytes within the shared memory allocated to the interleaver and deinterleaving the second plurality of RS coded data bytes within the shared memory allocated to the deinterleaver, wherein the shared memory allocated to the interleaver is used at the same time as the shared memory allocated to the deinterleaver.	89
vii.	Claim 2: The system of claim 1, wherein the determining is based on an impulse noise protection requirement.	92
viii.	Claim 3: The system of claim 1, wherein the determining is based on a latency requirement.	94
ix.	Claim 4: The system of claim 1, wherein the determining is based on a bit error requirement.	96

XIII.	Ground 2: The combination of VDSL1 and Fadavi-Ardekani renders the challenged claims obvious.	98
b.	Analysis	98
i.	Claim 1: [1.pre] A system that allocates shared memory comprising:	98
ii.	Claim 1: [1.A] a transceiver that is capable of: transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver;	106
iii.	Claim 1: [1.B] determining an amount of memory required by the interleaver to interleave a first plurality of Reed Solomon (RS) coded data bytes within the shared memory;	114
iv.	Claim 1: [1.C] allocating a first number of bytes of the shared memory to the interleaver to interleave the first plurality of Reed Solomon (RS) coded data bytes for transmission at a first data rate, wherein the allocated memory for the interleaver does not exceed the maximum number of bytes specified in the message;	117
v.	Claim 1: [1.D] allocating a second number of bytes of the shared memory to a deinterleaver to deinterleave a second	

	plurality of RS coded data bytes received at a second data rate; and	122
vi.	Claim 1: [1.E] interleaving the first plurality of RS coded data bytes within the shared memory allocated to the interleaver and deinterleaving the second plurality of RS coded data bytes within the shared memory allocated to the deinterleaver, wherein the shared memory allocated to the interleaver is used at the same time as the shared memory allocated to the deinterleaver.....	123
vii.	Claim 2: The system of claim 1, wherein the determining is based on an impulse noise protection requirement.	125
viii.	Claim 3: The system of claim 1, wherein the determining is based on a latency requirement.	128
ix.	Claim 4: The system of claim 1, wherein the determining is based on a bit error requirement.	130
XIV.	Revision or Supplementation	131

Exhibit	Short Name	Description
Ex. 1001	'048 Patent	U.S. Patent No. 8,276,048
Ex. 1002	'048 File History	File History of U.S. Patent No. 8,276,048
Ex. 1003	Wesel Declaration	Declaration of Dr. Richard Wesel, under 37 C.F.R. § 1.68
Ex. 1004	Wesel CV	<i>Curriculum Vitae</i> of Dr. Richard Wesel
Ex. 1005	Mazzoni	U.S. Patent No. 7,269,208
Ex. 1006	Fadavi-Ardekani	U.S. Patent No. 6,707,822
Ex. 1007	VDSL1	ETSI TS 101 270-2 V1.2.1 Technical Specification
Ex. 1008	Forney	G. David Forney, <i>Burst-Correcting Codes for the Classic Bursty Channel</i> , Vol. Com-19, No. 5, IEEE Transactions on Communications, 772 (1971)
Ex. 1009	Zogakis	T. Nicholas Zogakis, <i>A Coded and Shaped Discrete Multitone System</i> , Vol. 43, No. 12, IEEE Transactions on Communications, 2941 (1995)
Ex. 1010	Kernighan	B.W. Kernighan, D. M. Ritchie, <i>The C Programming Language</i> , 1998
Ex. 1011	AT&T DSP (1981)	J. R. Boddie, <i>Overview: The Device, Support Facilities, and Applications</i> , Vol. 60, The Bell System Technical Journal, 1431 (1981)
Ex. 1012	AT&T DSP 32 (1986)	J. R. Boddie, <i>The DSP32 Digital Signal Processor and its Application Development Tools</i> , AT&T Technical Journal, 89 (1986)
Ex. 1013	Texas Instruments TMS 320 (1994)	Mansoor A. Chishtie, <i>Telecommunications Applications With the TMS320C5x DSPs</i> , Digital Signal Processing Products (1994)

Exhibit	Short Name	Description
Ex. 1014	Maxwell	U.S. Patent No. 4,924,456
Ex. 1015	V.32 <i>bis</i>	ITU Recommendation V.32 <i>bis</i> (1991)
Ex. 1016	V.32	ITU Recommendation V.32 (1993)
Ex. 1017	LB-031	ITU Contribution LB-031 (June 2004)
Ex. 1018	Family 3 Claim Construction Opinion	<i>TQ Delta, LLC v. 2Wire, Inc.</i> , No. 13-cv-01835-RGA, ECF. No. 445 (D. Del. Dec. 18, 2017) (Claim Construction Opinion)
Ex. 1019	Family 3 Claim Construction Order	<i>TQ Delta, LLC v. 2Wire, Inc.</i> , No. 13-cv-01835-RGA, ECF. No. 454 (D. Del. Dec. 28, 2017) (Claim Construction Order)
Ex. 1020	Family 1 Claim Construction Opinion	<i>TQ Delta, LLC v. 2Wire, Inc.</i> , No. 13-cv-01835-RGA, ECF. No.477 (D. Del. Jan 30, 2018) (Claim Construction Opinion)
Ex. 1021	Family 1 Claim Construction Order	<i>TQ Delta, LLC v. 2Wire, Inc.</i> , No. 13-cv-01835-RGA, ECF. No. 484 (D. Del. Feb. 6, 2018) (Claim Construction Order)
Ex. 1022	Family 3 MSJ Opinion	<i>TQ Delta, LLC v. 2Wire, Inc.</i> , No. 13-cv-01835-RGA, ECF No. 1106 (D. Del. Apr. 25, 2019) (MSJ Opinion)
Ex. 1023	Family 3 Jury Form	<i>TQ Delta, LLC v. 2Wire, Inc.</i> , No. 13-cv-01835-RGA, ECF No. 1187 (D. Del. May 23, 2019) (Jury Form)
Ex. 1024	'890 Patent	U.S. Patent No. 7,831,890
Ex. 1025	'890 File History	File History of U.S. Patent No. 7,831,890
Ex. 1026	Nokia Complaint	<i>TQ Delta, v. Nokia Corp. et al.</i> , No. 2:21-cv-00309, Dkt. No. 1. (E.D. Tex. Aug. 13, 2021) (Nokia Complaint)

Exhibit	Short Name	Description
Ex. 1027	CommScope Complaint	<i>TQ Delta, LLC v. CommScope Holding Co. et al.</i> , No. 2:21-cv-00310, Dkt. No. 1 (E.D. Tex. Aug. 13, 2021) (CommScope Complaint)
Ex. 1028	Third and Final Scheduling Order	<i>TQ Delta, LLC v. 2Wire, Inc.</i> , No. 1:13-cv-01835, Dkt. No. 117 (D. Del. April 10, 2017) (Third and Final Scheduling Order)
Ex. 1029	2Wire Docket	<i>TQ Delta, LLC v. 2Wire, Inc.</i> , No. 1:13-cv-01835 (D. Del.) (Docket as of February 25, 2022)
Ex. 1030	Sept. 15, 2021 Letter	<i>TQ Delta, LLC v. 2Wire, Inc.</i> , No. 1:13-cv-01835, Dkt. No. 1591 (D. Del. Sept. 15, 2021) (Letter)
Ex. 1031	Docket Control Order	<i>TQ Delta, LLC v. CommScope Holding Co. et al.</i> , No. 2:21-cv-00310, Dkt. No. 62 (E.D. Tex. Dec. 17, 2021) (Docket Control Order)
Ex. 1032	Feb. 24, 2022 Letter Regarding ADTRAN Case Schedule	<i>TQ Delta, LLC v. ADTRAN, Inc.</i> , No. 1:14-cv-00954, Dkt. No. 1373 (Feb. 24, 2022) (Letter)
Ex. 1033	Hall-Ellis Declaration	Declaration of Sylvia Hall-Ellis
Ex. 1034	CommScope Motion to Transfer	<i>TQ Delta, LLC v. CommScope Holding Co. et al.</i> , No. 2:21-cv-00310, Dkt. No. 32 (E.D. Tex. Nov. 11, 2021) (CommScope Motion to Transfer)

DECLARATION OF RICHARD WESEL

I, Richard Wesel, do hereby declare and say as follows:

I. Introduction

1. My name is Dr. Richard Wesel, and I have been retained as a technical expert by counsel for Petitioner CommScope, Inc. (“CommScope”) to address certain issues regarding U.S. Patent No. 8,276,048 (“the ’048 Patent”).

2. I have personal knowledge of the facts contained in this Declaration, am of legal age, and am otherwise competent to testify.

3. Unless otherwise stated, the matters contained in this Declaration are of my own personal knowledge and, if called as a witness, I could and would testify completely and truthfully with regard to the matters set forth herein.

4. My opinions are based on my years of education, research, and experience, as well as my study of the relevant materials. A list of materials I considered is included at the beginning of this Declaration.

5. I reserve all available rights to supplement this Declaration if further information becomes available, or if I am asked to consider additional information. This Declaration represents only those opinions I have formed to date.

6. Further, I reserve all available rights to consider and comment on any additional expert statements or testimony of TQ Delta’s expert(s) in this matter.

7. I am being compensated at my usual consulting rate of \$600 per hour for my time spent working on issues in this matter. My compensation does not depend on the outcome of this matter or the opinions I express.

II. Qualifications

8. A true and correct copy of my curriculum vitae is attached as Ex. 1004 to this Declaration.

9. I have over 30 years of experience in communications and signal processing. I am a professor in the Electrical and Computer Engineering Department of UCLA as well as the Associate Dean for Academic and Student Affairs of the Henry Samueli School of Engineering and Applied Science. After receiving bachelors and master's degrees in Electrical Engineering from MIT in 1989, I worked at AT&T Bell Laboratories from 1989 to 1991 as a Member of Technical Staff performing research and development in telecommunications. I attended Stanford University from 1991 to 1996, receiving my Ph.D. in Electrical Engineering in 1996. Upon receiving my Ph.D. I joined the faculty of the UCLA Electrical Engineering Department, where I have been teaching and doing research on communications and signal processing for the last 24 years.

10. I have extensive experience researching and teaching communications techniques including multi-carrier modulation, interleaving, and error control coding in general as well as Reed-Solomon coding in particular. My publications

and patents are listed in my CV. I have published over 200 conference and journal publications, and I am the inventor on nine patents. I have received the National Science Foundation (NSF) CAREER Award and an Okawa Foundation award for research in information theory and telecommunications. I am a Fellow of the Institute of Electrical and Electronics Engineers (IEEE). I previously served as an Associate Editor for Coding and Coded Modulation for the IEEE Transactions on Communications. I currently serve as an Associate Editor for Coding and Decoding for the IEEE Transactions on Information Theory.

11. I authored an Asilomar conference paper in 1995 entitled “Fundamentals of Coding for Broadcast OFDM” that discusses techniques for designing coding and modulation for multicarrier transmission with interleaving. My 1996 Ph.D. dissertation included the design of trellis codes for multicarrier transmission with interleaving. In a 1999 Communications Letter, I presented related results on trellis codes for multicarrier transmission with interleaving. In a 2000 IEEE Transactions on Communications paper, I compared the performance of these new codes to Reed-Solomon codes used on multicarrier transmission with interleaving. I also authored several conference papers in the late 1990’s related to coding for multicarrier transmission with interleaving including “Joint Interleaver and Trellis Code Design,” “Periodic Symbol Puncturing of Trellis Codes,” and “Trellis Codes for Compound Periodic Gaussian Channels.”

12. I authored the chapter entitled “Error Control” in the book Wireless Multimedia Communications: Networking Video, Voice, and Data in 1997, which discussed the concepts of Reed-Solomon codes and interleaving. This book also discusses multicarrier modulation in Section 3.4 and Section 5.2.

13. I am an inventor on U.S. Patent No. 6,125,150 entitled “Transmission System using Code Designed for Transmission with Periodic Interleaving,” which discloses, among other things, error control techniques for multicarrier transmission with interleaving. I am also an inventor on U.S. Patent No. 6,158,041 filed in October 1998 entitled “System and method for I/Q Trellis Coded Modulation” which also describes, among other things, error correction coding techniques for multicarrier transmissions using interleaving.

III. Scope of the Engagement

14. I have been retained by Goodwin Procter LLP on CommScope's behalf to provide various analyses and opinions. Among other tasks, I have been asked to address the technology claimed in the '048 Patent and to evaluate whether certain claims of the '048 Patent would have been obvious to a person of ordinary skill in the art.

15. All of the opinions I express in this Declaration have been made from the standpoint of a person of ordinary skill in the relevant art at the time of the

invention of the '048 Patent, which I am informed is, for purposes of this matter, the earliest claimed priority date of October 12, 2004.

16. I have been asked to identify the level of skill in the art pertinent to the '048 Patent. It is my opinion that a person of ordinary skill in the art pertinent to the '048 Patent would have had at least a Bachelor's degree in electrical engineering, or a related field, and at least 6–7 years of experience in telecommunications or a related field; a master's degree in electrical or computer engineering, or the equivalent, and at least 4–5 years of experience in telecommunications or a related field; or a Ph.D. in electrical or computer engineering, or the equivalent, with at least 1–2 years of experience in telecommunications or a related field. A person with a different degree but with additional relevant experience could still qualify if the additional experience compensates for the different educational background.

17. I consider myself a person of ordinary skill in the art as of at least the earliest claimed priority date (October 12, 2004).

18. My opinions are based on my experience and knowledge and the information I have reviewed as of the date of this Declaration. In connection with my analysis, I have reviewed the exhibits listed in the above list of exhibits, as well as each of the items referenced herein. My opinions directed to the invalidity of the challenged claims are based, at least in part, on the following prior art publications:

- **“Mazzoni”** (Ex. 1005), which is US. Patent No. 7,269,208, filed on July 11, 2001 and issued on September 11, 2007;
- **“Fadavi-Ardekani”** (Ex. 1006), which is U.S. Patent No. 6,707,822, filed on January 7, 2000 and issued on March 16, 2004;
- **“VDSL1”** (Ex. 1007), which is technical specification ETSI TS 101 270-2 put forth by the European Telecommunications Standards Institute related to “Very High Speed Digital Subscriber Line (VDSL).” This technical specification was published online by ETSI on July 24, 2003. POSA’s at the time of this publication regularly looked to publications from standards organizations, including ETSI, for guidance. Thus, a POSA that was interested in VDSL implementation would have looked to ETSI for publications on technical specifications related to VDSL and would have found, and relied upon, this specification. *See* Ex. 1033.

IV. Legal Standards Relied Upon

19. For purposes of this Declaration, counsel for has instructed me to make the following assumptions:

20. Claims are construed from the perspective of a person of ordinary skill in the art (“POSA”) as of the effective filing date of the patent application. Persons of ordinary skill in the art are deemed to read the claims in the context of

the entire patent, including the specification and prosecution history. In other words, the terms are not considered in a vacuum.

21. It is my understanding that information that satisfies one of the categories of prior art set forth in 35 U.S.C. § 102 may be used in an invalidity analysis under §§ 102 or 103. If information is not properly classified as prior art under one of the subsections of § 102 of the Patent Act, then it may not form the basis of an anticipation or obviousness determination. It is also my understanding that, for inter partes review, applicable prior art is limited to patents and printed publications.

22. I am also informed and understand that a patent claim is invalid under 35 U.S.C. § 103 if the differences between the invention and the prior art are such that the subject matter as a whole would have been obvious at the time of the invention to a person having ordinary skill in the art to which the subject matter pertains. Obviousness, I am informed, is determined based on the scope and content of the prior art, the differences between the prior art and the claim, the level of ordinary skill in the art, and secondary indications of non-obviousness to the extent they exist.

23. I understand that whether there are any relevant differences between the prior art and the claimed invention is to be analyzed from the view of a person of ordinary skill in the art at the time of the invention. A person of ordinary skill in

the art is a hypothetical person who is presumed to be aware of all of the relevant art at the time of the invention. The person of ordinary skill is not an automaton, and may be able to fit together the teachings of multiple patents (or other printed publications) by employing ordinary creativity and the common sense that familiar items may have obvious uses in another context or beyond their primary purposes.

24. In analyzing the relevance of the differences between the claimed invention and the prior art, I understand that I must consider the impact, if any, of such differences on the obviousness or non-obviousness of the invention as a whole, not merely some portion of it. The person of ordinary skill faced with a problem is able to apply his or her experience and ability to solve the problem and also look to any available prior art to help solve the problem.

25. I understand that an invention would have been obvious if a person of ordinary skill in the art, facing the wide range of needs created by developments in the field, would have seen an obvious benefit to the solutions tried by the patent applicant. When there is a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, it would have been obvious to a person of ordinary skill to try the known options. If a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique would have been obvious.

26. I understand that I do not need to look for precise teachings in the prior art directed to the subject matter of the claimed invention. I understand that I may consider the inferences and creative steps that a person of ordinary skill in the art would have employed in reviewing the prior art at the time of the invention. For example, if the claimed invention combines elements that were known in the prior art and if the combination yields results that were predictable to a person of ordinary skill in the art at the time of the invention, then this would make it more likely that the claim was obvious. On the other hand, if the combination of known elements yields unexpected or unpredictable results, or if the prior art teaches away from combining the known elements, then this would make it more likely that the claim that successfully combined those elements was not obvious. I understand that hindsight must not be used when comparing the prior art to the invention during an obviousness analysis.

27. I understand that obviousness may be shown by demonstrating that it would have been obvious to modify what is taught in a single piece of prior art to create the patented invention. Obviousness may also be shown by demonstrating that it would have been obvious to combine the teachings of more than one item of prior art. I understand that a claimed invention may be obvious if some teaching, suggestion, or motivation exists that would have led a person of ordinary skill in the art to combine the invalidating references. I also understand that this suggestion

or motivation may come from sources such as explicit statements in the prior art, or from the knowledge of a person having ordinary skill in the art. Alternatively, any need or problem known in the field at the time and addressed by the patent may provide a reason for combining elements of the prior art. I also understand that when there is a design need or market pressure, and there are a finite number of predictable solutions, persons of ordinary skill may be motivated to apply both their skill and common sense in trying to combine the known options in order to solve the problem.

28. I understand that absent evidence of an invention date prior to the filing date of a patent, the invention date of the patent is presumed to be its filing date. A prior invention requires a complete conception of the invention and a reduction to practice of that invention. The patent owner has the burden of establishing by clear and convincing evidence a date of conception earlier than the filing date of the patent.

V. Technical Background

29. In this technical background, I describe the pertinent technology as it existed as of October 2004, the assumed priority date of the '048 Patent. As I will describe, the ideas of Reed-Solomon coding, interleaving, shared memory, and configuration messages that allow transceivers to exchange capabilities and agree on the parameters that define the communication link (including messages about

maximum available interleaver memory) were all known to a POSA at the time of the alleged invention.

a. Reed-Solomon Coding

30. Reed-Solomon codes are a particular type of error control coding, sometimes called forward error correction (FEC) coding. Error control coding protects a message by using additional bits or symbols called redundancy to produce a codeword that can withstand errors in the channel. The particular Reed-Solomon codes discussed in the patents at issue use 8-bit symbols, which are often called bytes or octets. A Reed Solomon code adds R extra symbols, sometimes called the R parity bytes or check bytes, to the K information symbols. The parameter R is referred to as the redundancy.

31. Reed-Solomon codes are a particularly powerful form of error correcting code because they achieve the Singleton bound. Such codes have the property that if the code has redundancy R , then the code will be able to correct any error pattern with $\left\lfloor \frac{R}{2} \right\rfloor$ symbols. For example, a Reed-Solomon code with a redundancy of 16 parity bytes can correct any error pattern with 8 or fewer byte errors. An example of a Reed-Solomon code with 16 parity bytes is the (240,224) Reed-Solomon code with 224 information bytes and 16 parity bytes for a total of 240 bytes in each transmitted codeword.

32. Reed-Solomon coding had long been a tool commonly used by a POSA at the time of the alleged invention. Reed-Solomon codes were discovered by Reed and Solomon in 1960 and are a special case of non-binary BCH codes that were discovered independently by Bose and Ray-Chaudhuri in 1960 and Hocquenghem in 1959.

b. Interleaving

33. Interleaving re-orders (permutes) symbols so that impairments, such as a burst of noise, that impact consecutive transmitted symbols will be spread out over multiple codewords. Each Reed-Solomon codeword has a limit on the number of byte errors that it can correct. With interleaving, a relatively long burst of errors can be spread over multiple Reed-Solomon codewords, allowing all the errors to be corrected.

34. One type of interleaver is the convolutional interleaver, and one implementation of a convolutional interleaver is the triangular implementation.

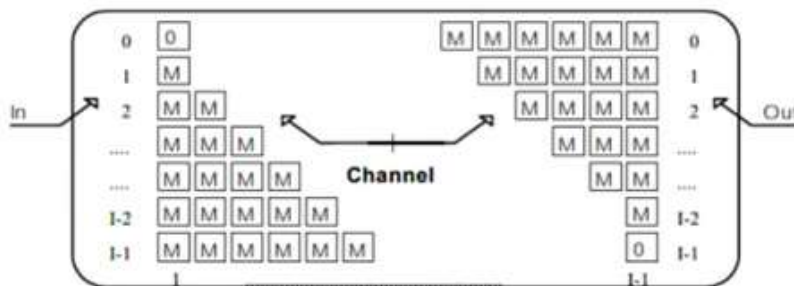


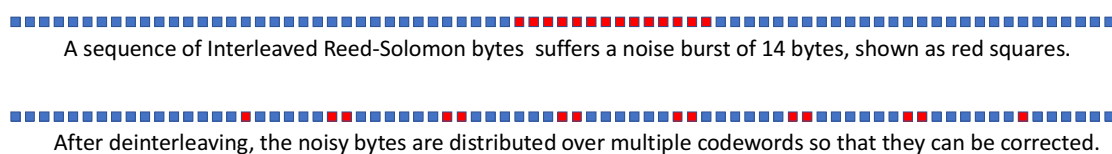
Figure B.2: Interleaver/De-interleaver Implementation example

35. Figure B.2 from VDSL1, VDSL1 at 153, shows the triangular implementation of an interleaver with parameters I and M . Each box containing the letter M represents M memory elements used to delay bytes written into the interleaver. Figure B.2 shows I rows corresponding to delay lines or shift registers (or virtual shift registers (*See* VDSL1 at 152-153.) implemented, for example, in a random access memory), with indices $0, \dots, I-1$. At the transmitting transceiver, I Reed-Solomon bytes at a time are written into the interleaver at the left side of each shift register. Then, I bytes are read from the interleaver at the right side of each shift register and are transmitted. Bytes written to the top shift register experience a delay of zero bytes before transmission. Bytes written to the next shift register experience a delay of M bytes. Bytes written to the shift register after that experience a delay of $2M$ bytes and so on. The bottom shift register is delayed by $(I-1) \times M$ bytes before transmission.

36. To undo the interleaving, the deinterleaver also uses I shift registers, with indices $0, \dots, I-1$, which are matched to the shift registers of the interleaver. The top shift register of the deinterleaver will delay its inputs by $(I-1) \times M$ bytes, the next shift register delays inputs by $(I-2) \times M$ bytes, and so on. In the deinterleaver, the bottom shift register implements a delay of zero bytes. In this way, the overall effect of the combination of the interleaver and deinterleaver is to delay each column by $(I-1) \times M$ bytes. Considering that each of the I bytes in a column is

transmitted separately, the overall delay experienced by each byte is exactly $I \times (I-1) \times M$. Thus, the deinterleaver returns the interleaved sequence back to its original order, but the combination of interleaving and deinterleaving delays the sequence by $I \times (I-1) \times M$ bytes.

37. To see how a convolutional interleaver spreads a noise burst so that it can be successfully decoded by Reed-Solomon codewords, consider the following simple example. Let's suppose our codewords have length $N = 8$ and that each codeword can correct two errored bytes. Without interleaving, fourteen consecutive byte errors would certainly cause a decoding error for a sequence of codewords that can each only correct two-byte errors. However, the figure below shows how the convolutional deinterleaver shown in Figure B.2 with $I = 7$ and $M = 1$ separates the error burst into isolated pairs of byte errors, where each pair of errors will be corrected by a different codeword.



38. Figure B.2 shows how the parameters I and M precisely determine the amount of memory that needs to be allocated for the triangular implementation of a convolutional interleaver. Each square containing the letter M represents M memory elements used to delay bytes written into the interleaver. Counting up the

boxes, there are $I-1$ in the first column, $I-2$ in the second column, and so on until the last column, which has only one box containing M memory elements. Since the sum of $(I-1) + (I-2) + (I-3) + \dots + 1 = \frac{I \times (I-1)}{2}$, the total amount of memory that we need to allocate for the interleaver is precisely $M \times \frac{I \times (I-1)}{2}$. The same argument shows that the associated deinterleaver with parameters I and M also requires an allocation of $M \times \frac{I \times (I-1)}{2}$ bytes of memory for the specific triangular implementation.

39. The concepts of interleaving in general and the interleaving of Reed-Solomon data bytes specifically were well known to a POSA long before the alleged invention. For example, in his 1971 journal paper in the *IEEE Transactions on Communications*, entitled “Burst-Correcting Codes for the Classic Bursty Channel,” G. David Forney discussed using convolutional interleaving with error control coding as a way to overcome bursts of noise. Ex. 1008.

40. Certainly by 1995, the combined use of Reed-Solomon coding and interleaving was well known. It was used, for example in in the multicarrier transmission system described in the *IEEE Transactions on Communications* article, “A Coded and Shaped Discrete Multitone System” by Zogakis and Aslanis published in December of 1995. Ex. 1009. Figure 9 of that article shows an interleaver processing the Reed-Solomon encoded bytes. Ex. 1009 at 2948.

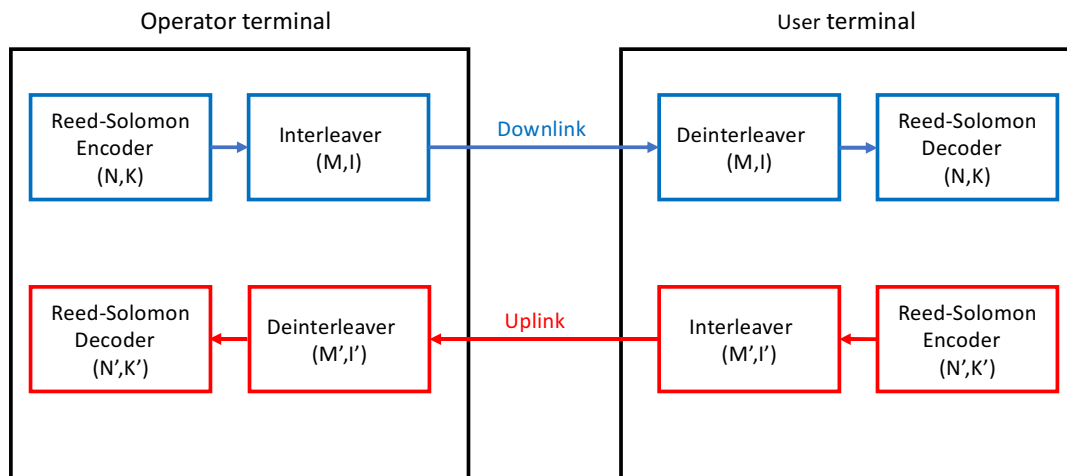
c. Shared Memory

41. Computing devices have long utilized common memory used by at least two functions. In fact, to the extent shared memory is construed more narrowly, computing devices have also long utilized common memory used by at least two functions, where a portion of the memory can be used by either one of the functions. The standard libraries defined for the C programming language include functions such as malloc and calloc that obtain blocks of memory dynamically. *See* Ex. 1010. Furthermore, standard communications functions including Reed-Solomon coding and interleaving were implemented on digital signal processors (DSPs) such as the AT&T DSP (1981), the AT&T DSP 32 (1986), and the Texas Instruments TMS 320 (1994). *See* Exs. 1011–1013. These digital signal processing chips all featured common memory used by at least two functions, where a portion of the memory can be used by either one of the functions.

d. Configuration Messages

42. I understand that a POSA at the time of the invention would have understood that whenever two transceivers have multiple communication configurations from which to choose, the transceivers must have a way to learn each other's capabilities and to agree on the specific configuration that they will use to communicate.

43. Consider the system below with a transceiver at an operator terminal communicating with a transceiver at a user terminal. The communication channel from the operator to the user is called the downlink, and the channel from the user to the operator is called the uplink. In this example system, as in many prior art systems, including ADSL and VDSL1, Reed-Solomon coding and interleaving are used for both the uplink channel and the downlink channel. A POSA would have understood that for the communication link to function properly, the Reed-Solomon decoder and convolutional deinterleaver in the user transceiver for the downlink must use the exact same parameters (N,K,M,I) as the Reed-Solomon encoder and interleaver for the downlink in the operator terminal. Similarly, the operator and user must agree on parameters (N',K',M',I') for the Reed-Solomon coding and interleaving for the uplink.



44. A POSA would have known that a common solution is to use configuration messages to exchange capabilities and agree on parameters. Such configuration messages are used in every ADSL standard as well as in the VDSL1 standard. As another example, in U.S. Patent No. 4,924,456 (Ex. 1014), which issued on May 8, 1990, it is explained that a modem uses configuration messages in an “Interactive state . . . to control its operation or to configure its operating characteristics. Ex. 1014 at 11:30–32. Likewise, the V.32 bis (Ex. 1015), and V.32 (Ex. 1016) bis modem standards explicitly handle transceivers with different capabilities since a V.32 bis modem communicating with an older V.32 modem will learn of the limited rate capabilities of the older modem through configuration messages and choose a rate that both modems can implement.

45. Such knowledge is in line with the implementation of the VDSL1 standard, which published in July 2003, and discloses communicating the capabilities of the transceiver in a configuration message, including by delivering a message specifying the maximum available interleaver memory. *See, e.g.*, Ex. 1007 at 132. In fact, Contribution LB-31, an ITU-T contribution that followed the VDSL1 standard, acknowledges that one known option for communicating interleaver capabilities was to specify in a message “an amount of memory.”. *See* Ex. 1017 at 6.

VI. The '048 Patent

46. The '048 Patent relates to sharing a common memory and allocating that common memory between an interleaver and a deinterleaver in digital subscriber line (“DSL”) technology. Ex. 1001 at Abstract.

47. I understand the filing date of the Provisional Application No. 60/618,269, to which the '048 Patent claims priority, is October 12, 2004.

48. The claims of the '048 Patent that are being challenged and that I have been asked to evaluate are claims 1–4. I have reproduced the challenged claims below:

Claim	Recitation
[1.Pre]	A system that allocates shared memory comprising:
[1.A]	a transceiver that is capable of: transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver;
[1.B]	determining an amount of memory required by the interleaver to interleave a first plurality of Reed Solomon (RS) coded data bytes within the shared memory;
[1.C]	allocating a first number of bytes of the shared memory to the interleaver to interleave the first plurality of Reed Solomon (RS) coded data bytes for transmission at a first data rate, wherein the allocated memory for the interleaver does not exceed the maximum number of bytes specified in the message;
[1.D]	allocating a second number of bytes of the shared memory to a deinterleaver to deinterleave a second plurality of RS coded data bytes received at a second data rate;
[1.E]	and interleaving the first plurality of RS coded data bytes within the shared memory allocated to the interleaver and deinterleaving the second plurality of RS coded data bytes within the shared memory allocated to the deinterleaver, wherein the shared memory allocated to the interleaver is used at the same time as the shared memory allocated to the deinterleaver.
2	The system of claim 1, wherein the determining is based on an impulse noise protection requirement.
3	The system of claim 1, wherein the determining is based on a latency requirement.
4	The system of claim 1, wherein the determining is based on a bit error rate requirement.

VII. Prosecution History

49. U.S. Patent Application No. 12/901,699 was originally filed on October 11, 2010. Ex. 1001. After a non-final rejection by the examiner on January 6, 2012, Patent Owner made small modifications to the claims and filed terminal

disclaimers on July 5, 2012. Ex. 1002 at 133–499. A Notice of Allowance was mailed shortly after, on July 27, 2012. *Id.* at 91–100. The Examiner explained that the “prior arts of record, namely Fadavi-Ardekani et al (US-6707822), teach sharing a memory between the interleavers and deinterleavers of multiple Asymmetrical Digital Subscriber Line (ADSL) sessions.” *Id.* at 98. The Examiner continued by stating that the prior art, however, “fail to teach, singly or in combination, the claimed invention as a whole, especially transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver.” *Id.* (emphasis in original).

50. The '048 Patent is a continuation of U.S. Patent Application 12/761,586, granted as U.S. Patent No. 7,844,882, which was allowed shortly after filing over Fadavi-Ardekani because, while Fadavi-Ardekani “discloses sharing a memory between the interleavers and deinterleavers of multiple ADSL sessions,” it failed to disclose “the distinct features of specifying a maximum number of bytes available to be allocated to [a/an] [deinterleaver/interleaver] in a transmitted or received message, determining the amount of memory required by the [deinterleaver/interleaver], and then allocating the bytes of shared memory to the [deinterleaver/interleaver] wherein the bytes allocated do not exceed the maximum number of available bytes specified in the message.” Ex. 1034 at 82, 83. The '882 Patent is a continuation of U.S. Patent Application 11/246,163, granted as U.S.

Patent No. 7,831,890 (Ex. 1024), which had all of its claims initially rejected under §102 in view of Fadavi-Ardekani. Ex. 1025, 317-33. After canceling all rejected claims, Patent Owner presented new claims, which were allowed for the same reasons the claims of the '882 Patent were allowed. *Id.*, 74-82.

VIII. Prior Decisions by the District Court

51. The '048 Patent was previously considered by the United States District Court for the District of Delaware in *TQ Delta, LLC v. 2Wire, Inc.*, No. 13-cv-01835-RGA (D. Del.) (“2Wire”); *TQ Delta, LLC v. Zhone Technologies, Inc.*, No. 13-cv-01836-RGA (D. Del.); *TQ Delta, LLC v. ZyXEL Communications, Inc.*, No. 13-cv-02013-RGA (D. Del.); and *TQ Delta, LLC v. ADTRAN, Inc.*, No. 14-cv-00954-RGA (D. Del.). These four cases were consolidated into lead case *2Wire*.¹

52. The district court issued memoranda and orders construing multiple claim terms of the '048 Patent. Exs. 1018-1021. The district court also addressed, on summary judgment, the invalidity of claim 1 of the '048 Patent as obvious in view of Mazzoni (Ex. 1005) over LB-031 (Ex. 1017), ultimately denying the motion because there was a dispute of material fact regarding the motivation to

¹ While not presently asserted against Nokia, the '048 Patent has also been asserted in the Eastern District of Texas against CommScope. *TQ Delta, LLC v. CommScope Holding Company, Inc.*, No. 2-21-cv-00310 (EDTX) (“CommScope”).

combine. Ex. 1022. Patent Owner also alleged certain references, including Fadavi-Ardekani (Ex. 1006), failed to disclose a “shared memory.” The court granted Patent Owner’s motion for no invalidity on this ground as a matter of law, reasoning that the Defendant failed to rebut Plaintiff’s argument that Defendant had not proven inherency. *Id.* at 18.

53. A jury subsequently determined that LB-031 as understood by a POSA and LB-031 in combination with Mazzoni did not render claim 1 of the ’048 Patent invalid as obvious. Ex. 1023. No ground on the basis of Mazzoni in view of VDSL1, or VDSL1 in view of Fadavi-Ardekani was presented to the jury. No dispositive motions related to the ’048 Patent have been filed in the ADTRAN litigation, and no jury trial has been scheduled.

IX. Litigation Related to the ’048 Patent

54. I understand that the ’048 Patent is at issue in one ongoing litigation, which does not involve the Petitioner.

X. Claim Construction

55. I understand that the in the *2Wire* litigation, terms of the ’048 Patent were construed. *See* Exs. 1018–21. The table below provides these constructions.

Claim Term	Delaware Construction
“shared memory”	“common memory used by at least two functions, where a portion of the memory can be used by either one of the functions”

“amount of memory”	Plain meaning, i.e., that is not limited to number of bytes
“the shared memory allocated to the [deinterleaver / interleaver] is used at the same time as the shared memory allocated to the [interleaver/ deinterleaver]”	“the deinterleaver reads from, writes to, or holds information for deinterleaving in its respective allocation of the shared memory at the same time as the interleaver reads from, writes to, or holds information for interleaving in its respective allocation of the shared memory”
“transceiver”	“communications device capable of transmitting and receiving data wherein the transmitter portion and receiver portion share at least some common circuitry”

a. “transceiver”

56. All of the challenged claims contain language claiming a “transceiver.” The Delaware district court previously construed transceiver to mean “communications device capable of transmitting and receiving data wherein the transmitter portion and receiver portion share at least some common circuitry.” In my opinion, this construction too narrowly defines “transceiver,” and rather the term should be given its plain and ordinary meaning, which is a “communications device capable of transmitting and receiving data.”

b. “shared memory”

57. All of the challenged claims contain language claiming a “shared memory.” The Delaware district court previously construed shared memory to mean “common memory used by at least two functions, where a portion of the memory can be used by either one of the functions.” In my opinion, this

construction too narrowly defines “shared memory,” and rather the term should be given its plain and ordinary meaning.

c. “amount of memory”

58. All of the challenged claims contain language claiming an “amount of memory.” The Delaware district court previously construed amount of memory as the plain meaning, i.e., “that is not limited to number of bytes.” In my opinion, this claim language should be construed simply as its plain and ordinary meaning.

d. “the shared memory allocated to the [deinterleaver / interleaver] is used at the same time as the shared memory allocated to the [interleaver/ deinterleaver]”

59. All of the challenged claims contain language claiming an “the shared memory allocated to the [deinterleaver / interleaver] is used at the same time as the shared memory allocated to the [interleaver/ deinterleaver].” The Delaware district court previously construed this term as “the deinterleaver reads from, writes to, or holds information for deinterleaving in its respective allocation of the shared memory at the same time as the interleaver reads from, writes to, or holds information for interleaving in its respective allocation of the shared memory.” This construction is consistent with my interpretation of this phrase.

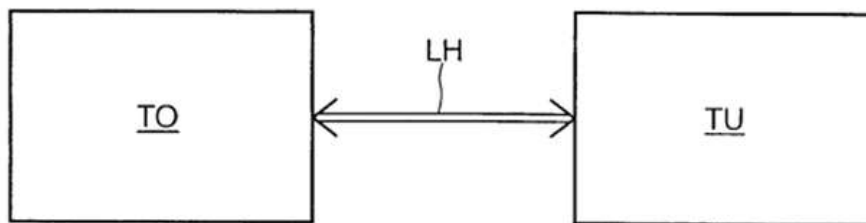
XI. Prior Art References

e. Mazzoni Overview

60. Mazzoni, which issued on September 11, 2007, derives from U.S. Patent Application No. 10/088,387, which was filed on July 11, 2001, and published as US2003/0021338 on January 30, 2003.

61. Mazzoni teaches a transceiver in a VDSL environment that can send and receive digital data and that can determine the shared memory capacity used to interleave and deinterleave that data. Ex. 1005 at 1:10–15. Specifically, as shown in Figure 1 below, Mazzoni contemplates two send/receive devices (i.e., transceivers), TU and TO, that communicate by “a very high bit rate communication line LH.” *Id.* at 1:60–61.

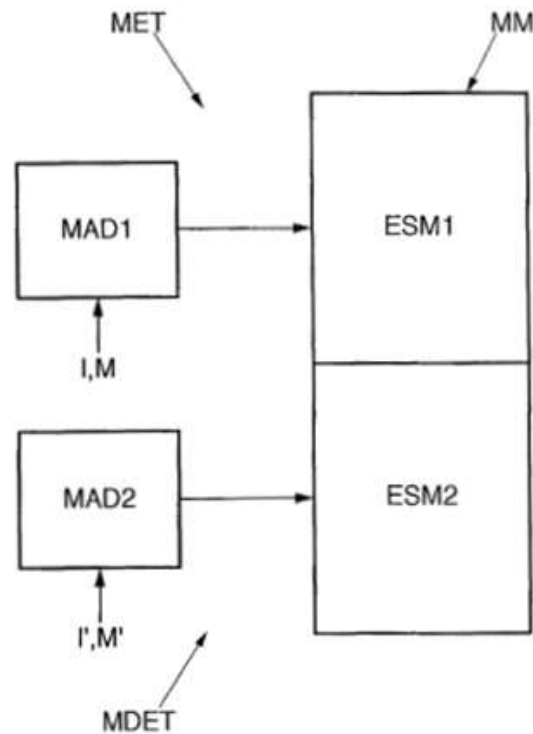
FIG.1



62. Mazzoni further discloses a shared memory “MM” that resides in each transceiver, and that “can be shared between the interleaving means and the deinterleaving means.” *Id.* at 1:59–65; 6:6–8. For example, Figure 6 of Mazzoni, shown below, illustrates how the common memory, MM, is allocated to the

interleaver (MET) as ESM1 and deinterleaver (MDET) as ESM2 on just one of the transceivers TO or TU.

FIG.6



63. Mazzoni explicitly teaches how to determine the required memory allocations for the interleaver and deinterleaver (i.e., the memory size of ESM1 and ESM2) via triangular implementation of convolutional interleaving and deinterleaving. Mazzoni further teaches using Reed-Solomon coding coupled to an interleaver to protect transmitted data. *Id.* at 2:25–30.

64. Thus, starting with downstream and upstream bit rates between the TO and TU that are disturbed by an impulsive noise, Mazzoni discloses calculating the number of data bits that are affected by an impulsive noise, the resultant errored bytes, the number of Reed-Solomon codewords needed to correct those errored bytes, and the required interleaver or deinterleaver memory to hold those bytes. *Id.* at 6:11–36.

65. Relying on those calculations, Mazzoni also discloses determining the amount of memory required by the interleaver/deinterleaver to interleave/deinterleave the Reed Solomon coded data bytes using parameters I and M. *Id.* at 6:37–38. Mazzoni explains that parameters I and M can be applied to the equation $I \times (I-1) \times M / 2$ to calculate “the size of the first memory space [ESM1]” for the interleaver, and parameters I’ and M’ can then be applied to the equation $I' \times (I'-1) \times M' / 2$ to calculate “the size of the second memory space ESM2” for the deinterleaver. *Id.* at 6:38–44. Mazzoni explains that the common memory required (MM) is the sum of ESM1 and ESM2.

f. Fadavi-Ardekani Overview

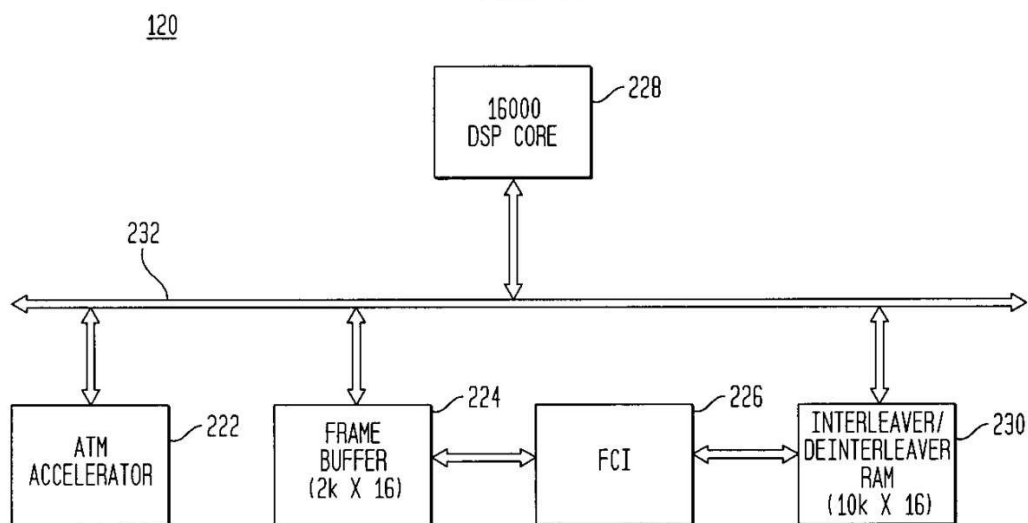
66. Fadavi-Ardekani was filed on January 7, 2000 and issued as a U.S. Patent on March 16, 2004.

67. Fadavi-Ardekani teaches a transceiver that may be implemented in a VDSL communication system with an interleaver and deinterleaver utilizing a

shared memory. Ex. 1006 at 4:19–21. Fadavi-Ardekani discloses a “Framer/Coder/Interleaver” (FCI), which “performs various...including...Reed-Solomon encoding/decoding, and interleaving/de-interleaving.” *Id.* at 6:11–16. The memory that supports the FCI is the Interleave/De-Interleave Memory (IDIM), which “allow[s] beneficial performance of various interleave/de-interleave processes.” *Id.* at 6:55–65.

68. Fadavi-Ardekani also teaches that the IDIM may be a shared memory between the interleaver and the deinterleaver. *Id.* at 6:55–58 (The “Interleave/De-Interleave Memory (IDIM)...may be utilized in a ping-pang fashion.” Fadavi-Ardekani defines “ping-pang” to mean that “areas of the memory buffer are alternately utilized exclusively by one agent (a transceiver component for performing some function) and then by a second agent” and that “[a]s one area of memory is being used by a first agent, another area of memory can be used by a different agent.” *Id.* at 5:60–65. For example, Fadavi-Ardekani provides Figure 2, which identifies the interleaver and deinterleaver memory existing on a single RAM.

FIG. 2



69. Fadavi-Ardekani teaches two implementations of shared memory: (1) a simple implementation where the memory has enough capacity to implement a session without using some bytes for both interleaving and deinterleaving for that session; and (2) an optimal implementation where the RAM is not large enough to execute a session without using at least some bytes for both interleaving and deinterleaving even for that particular session. *Id.* at 7:25–33. Both the simple and optimal implementations have the property that a portion of the memory can be used by either one of the functions of interleaving or deinterleaving.

g. VDSL1 Overview

70. VDSL1 was published on July 24, 2003. VDSL1 is an ETSI technical specification produced by “ETSI Technical Committee Transmission and

Multiplexing (TM)” that “specifies requirements for transceivers that provide very high bit-rate digital transmission,” or VDSL. Ex. 1007 at 9. VDSL1 teaches incorporating VDSL onto transceivers VTU-O and VTU-R, and then determining the memory allocation for the interleaver and deinterleaver on each transceiver via triangular implementation of convolutional interleaving and deinterleaving. Ex. 1007 at 13, 59–60.

71. Specifically, VDSL1 teaches that “[i]nterleaving shall be used to protect the data against bursts of errors by spreading the errors over a number of Reed-Solomon codewords.” *Id.* at 59. To determine the memory required to interleave or deinterleave those bytes, VDSL1 teaches parameters I and M, which can be used to determine the “(De)interleaver memory size” with the equation $M \times I \times (I-1)/2$ (see Table 19 below). *Id.* at 60.

Table 19: Characteristics of triangular, convolutional interleaver

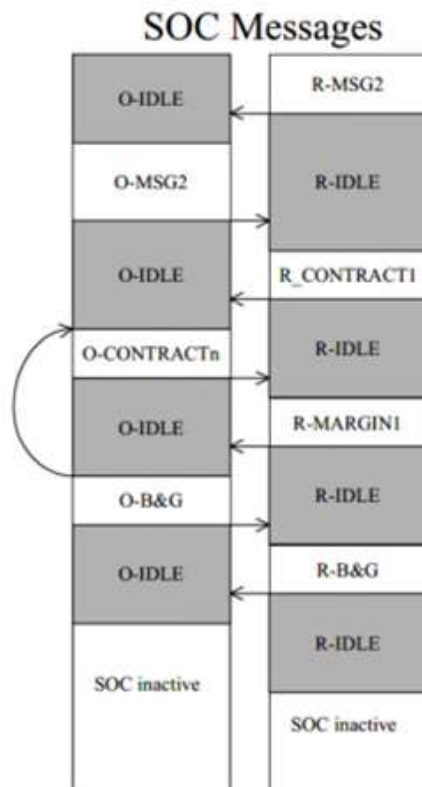
Parameter	Value
Interleaver block length	I octets (I must divide N)
Interleaving depth D	$M \times I + 1$ octets
(De)interleaver memory size	$M \times I \times (I-1)/2$ octets
End-to-end delay	$M \times I \times (I-1)$ octets
Correction capability	$\lfloor I/q \rfloor \times M \times (I+1)$ octets

72. VDSL1 then contemplates various “Interleaver Characteristics” in Table 27, including “Interleaver Memory Size,” “Erasure Correction (E),” and “End-to-End Delay (DL).” *Id.* at 72.

Table 27: Interleaver Characteristics

Parameter	Value	Notes
Block Length (I)	$I = N/8, N/4 \text{ or } N/2, \text{ octets}$	$N = PS+19, \text{ octets}$
Depth (D)	$D = M \times I + 1, \text{ octets}$	$M = 0 \rightarrow 64, \text{ programmable}$
Erasure Correction (E)	$E = \lfloor t \times I / N \rfloor \times (M \times I + 1), \text{ octets}$	$t = 8 \text{ (RS error correction ability)}$
End-to-End Delay (DL)	$DL = M \times I \times (I - 1), \text{ octets}$	
Interleaver Memory Size	$MEM = M \times I \times (I - 1) / 2, \text{ octets}$	
NOTE 1: The interleaver erasure correction E defines the maximum number of corrupted sequential octets could be corrected by RS algorithm when interleaving is applied. Correspondingly, the duration of noise pulses the system is protected from could be calculated as $E \times 8/R$, where R is the bit rate of the transmit signal over the line.		
NOTE 2: The $\lfloor \rfloor$ symbols indicates the truncation of the value to the lower integer.		

73. VDSL1 also teaches initialization protocols between the VTU-O and VTU-R to set those I and M parameters, including the messages shown below. *Id.* at 127.



74. The first message, R-MSG2, describes the VTU-R's capabilities, including the maximal number of bytes available to be allocated to an interleaver. *Id.* at 132. After the messages R-MSG2, O-MSG2, and R-CONTRACT1, the message O-CONTRACTn, contains a proposed upstream contract and a proposed downstream contract, including contract descriptors for each contract, including the parameters I and M for the upstream interleaver/deinterleaver and a distinct set of parameters I and M for the downstream deinterleaver/interleaver. *Id.* at 129.

h. Motivation to Combine the Prior Art

i. Motivation to Combine Mazzoni and VDSL1

75. Both Mazzoni and VDSL1 are directed to VDSL systems. Indeed, Mazzoni contemplates a transceiver for sending/receiving digital data, "in particular in a VDSL environment," Ex. 1005, and VDSL1 is the technical specification put forth by the European Telecommunications Standards Institute (ETSI) that "specifies requirements for transceivers that provide very high bit-rate digital transmission," i.e., VDSL, Ex. 1007 at 9.

76. As explained *supra*, Mazzoni teaches a VDSL transceiver with a shared memory that determines an amount of memory required by an interleaver/deinterleaver to interleave/deinterleave Reed Solomon coded data bytes, that allocates those bytes of the shared memory to the

interleaver/deinterleaver at a particular data rate, and that subsequently interleaves/deinterleaves those bytes.

77. Mazzoni does not, however, expressly disclose transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver or deinterleaver. A POSA would have understood that a transceiver as contemplated in Mazzoni, especially one implemented in a VDSL environment, would perform initialization protocols. Thus, a POSA would have been motivated to look to the VDSL technical specifications available around the same time for those initialization protocols—here, the VDSL1 technical specification.

78. VDSL1 disclosed initialization. Ex. 1007 at 111 (disclosing “Initialisation” protocols for the transceiver pair VTU-O and VTU-R, including an “[e]xchange of parameters”); id. at 127–34 (Disclosing a “channel analysis and exchange” state, which discloses the negotiation of interleaver/deinterleaver parameters, including R-MSG2, which specifies a maximum number of bytes of memory available to be allocated to an interleaver, and O-CONTRACT_n, which describes the specific parameters I and M for each interleaver/deinterleaver pair. Once the O-CONTRACT_n message is confirmed by transmission of the O-B&G message, these parameters indicate the agreed-upon memory allocation between

interleaving and deinterleaving in the context of the triangular implementation of interleaving taught by VDSL1.).

79. A POSA thus would have been motivated to combine the initialization protocols of VDSL1 with the transceivers disclosed by Mazzoni using the parameter exchange protocol of the VDSL1 technical specification so as to perform initialization in a VDSL environment. This simple incorporation would have achieved the known benefits of allowing the TO and TU transceivers of Mazzoni to communicate hardware capabilities and negotiate selection of parameters, thus allowing for increased flexibility in parameter selection and confirmation that the hardware is compatible with the executed parameters (e.g., that memory MM has enough storage capacity for the determined interleaver and deinterleaver size). *See, e.g.*, Ex. 1007 at 111.

80. There are several reasons to combine Mazzoni and VDSL1. First, Mazzoni expressly instructs that the invention “may advantageously be applied to a very high rate digital subscriber line (VDSL) environment or system.” Ex. 1005 at 1:19–21. Thus, the POSA would have been motivated to combine Mazzoni’s transceiver with a reference that provided the relevant implementation details that would have allowed it to operate in a VDSL environment—the technical specification. At the relevant time, a POSA looking to build a functioning

transceiver that operated in a VDSL environment would have sought to comply with the VDSL1 technical specification.

81. Second, the POSA would have understood that a VDSL transceiver would have had to perform initialization for the purposes of exchanging relevant information between the office transceiver and the remote transceiver. For example, VDSL1 explains that one of the initialization tasks between the office and remote transceiver is defining “a common mode of operation,” without which the office and remote transceivers would not be able to properly function. Ex. 1007 at 111. As a result, the POSA would have been motivated to refer to the standard so as to follow the common and required mechanism for sharing such information between the office and remote. Indeed, both Mazzoni and VDSL1 disclose additional requirements that would have been established during initialization, including impulse noise requirements, *compare* Ex. 1005 at 6:11–18, *with* Ex. 1007 at 71, latency requirements, *compare* Ex. 1005 at 2:54–56, *with* Ex. 1007 at 59, and bit error rate requirements, *compare* Ex. 1005 at 4:53–56, *with* Ex. 1007 at 72.

82. Third, the disclosure of Mazzoni explicitly suggests initialization protocols. For example, in VDSL1, the parameters I, M, I', and M' result from the negotiation between the VTU-R and VTU-O transceivers during the “Channel analysis and exchange” phase, i.e., during initialization. Ex. 1007 at 129.

Similarly, Mazzoni explains that “the parameters I, M, I’, and M’ can be modified, e.g., by software, and are delivered by control means MCD.” Ex. 1005 at 5:21–24. A POSA would have understood that initialization protocols would have preceded the delivery of M and I parameters by the MCD. In reading Mazzoni, a POSA thus would have been motivated to look to the VDSL1 technical specification and had a reasonable expectation of success in doing so. It is not significant that Mazzoni also discloses reading from a table of values for the parameters I, M, I’, and M’. *See* Ex. 1005 at 51–61. A POSA would have understood that initialization protocols could instruct which I, M, I’, and M’ parameters to select from said table, which would still provide the benefits of using initialization protocols as described above. Alternatively, a POSA would have understood that initialization protocols could substitute for such a table—which is merely one contemplated method of delivering the I and M parameters—to allow for the benefits described above.

83. In addition, because both Mazzoni and VDSL1 contemplate the exact same implementation of interleaving and deinterleaving—triangular implementation of convolutional interleaving—a POSA would have had a reasonable expectation of success in combining the two references. *Compare* Ex. 1005 at 2:49–56, *with* Ex. 1007 at 60, 152–53. Particularly, both Mazzoni and VDSL1 contemplate the same parameters to define the interleaver and deinterleaver—I, M, I’, and M’. *Compare* Ex. 1005 at 6:37–44, *with* Ex. 1007 at

59–60, 152–53. Likewise, both explain that to determine the interleaver or deinterleaver memory size—a critical element of the challenged claims—the parameters can be applied to the equation $I \times (I-1) \times M/2$.” Compare Ex. 1005 at 6:38–41, with Ex. 1007 at 60. A POSA using ordinary creativity would know to implement a memory in the transceiver of Mazzoni of a size sufficient to enable the range of parameters contemplated in VDSL1. Indeed, Mazzoni expressly instructs a POSA that its example is simply “[a]n application of the invention to a VDSL communication system,” and that “the invention is not limited to this application.” Ex. 1005 at 3:53–55.

84. For at least these reasons, in reading Mazzoni, a POSA thus would have been motivated to look to the VDSL1 standard and had a reasonable expectation of success in doing so.

ii. Motivation to Combine VDSL1 and Fadavi-Ardekani

85. Both VDSL1 and Fadavi-Ardekani are directed to VDSL systems. Ex. 1007 at Cover; Ex. 1006 at 4:19-21. Indeed, VDSL1 is the technical specification put forth by the European Telecommunications Standards Institute (ETSI) that “specifies requirements for transceivers that provide very high bit-rate digital transmission,” i.e., VDSL, and Fadavi-Ardekani contemplates that the transceiver of the invention may “incorporate other variations of DSL (i.e., xDSL),

such as High bit-rate DSL (HDSL) and Very high bit-rate DSL (VDSL).” Ex. 1007 at 9; Ex. 1006 at 4:19–21.

86. As described in Ground 1, VDSL1 is a standards document that includes, among other things, initialization instructions. Ex. 1007 at 111 (disclosing “Initialisation” protocols for the transceiver pair VTU-O and VTU-R, including an “[e]xchange of parameters”); *id.* at 127–34 (disclosing a “channel analysis and exchange” state, which discloses the negotiation of interleaver/deinterleaver parameters, including R-MSG2, which specifies a maximum number of bytes of memory available to be allocated to an interleaver). In so doing, VSL1 discloses various elements, such as a communication device (e.g., a transceiver). Ex. 1007 at 13.

87. VDSL1 does not, however, expressly disclose use of a shared memory between the interleaver and deinterleaver. A POSA would have understood that a transceiver as contemplated in VDSL1 would have implemented any efficiencies to meet those disclosed parameters. Here, implementing a shared memory was one known efficiency.

88. Fadavi-Ardekani disclosed shared memory between an interleaver and deinterleaver. Ex. 1006 at 5:60–65; 6:55–58 (“The Interleave/De-Interleave Memory (IDIM) 230 provides a memory” and “may be utilized in a ping-pang

fashion,” meaning that “[a]s one area of memory is being used by a first agent, another area of memory can be used by a different agent”).

89. A POSA thus would have been motivated to combine the shared memory of Fadavi-Ardekani with the transceivers disclosed by VDSL1 to create transceivers that are more efficient, which would provide the advantages of allowing each transceiver to support more sessions or sessions that are more data intensive and, thus, reduce costs. Ex. 1006 at 8:4–17 (explaining that the memory savings of the invention reduces cost). This simple substitution would have achieved the known benefits of allowing the VDSL1 transceiver to implement a wider range of interleaver and deinterleaver parameters for a given size of memory or to implement more sessions for given size of memory or to support a given range of interleaver and deinterleaver parameters with less memory.

90. There are several reasons to implement Fadavi-Ardekani’s shared memory in the system disclosed by VDSL1. First, VDSL1 describes determining the memory allocations for the interleaver and deinterleaver when using the triangular implementation of convolutional interleaving and deinterleaving but does not provide explicit instructions for the hardware that should or must be used to implement the standard. Fadavi-Ardekani, granted as a patent around the same time that the VDSL1 technical specification was published, provides the requisite hardware and instructs that its disclosure may beneficially be implemented with

VDSL. Ex. 1006 at 4:18–21. As a result, the POSA would have been motivated to combine a reference that provides increased efficiencies to the memory of the transceiver in a VDSL environment with VDSL1’s transceiver.

91. Second, the POSA would have understood that implementing the VDSL1 standard supporting *both* asymmetric and symmetric downstream and upstream data streams for the numerous sessions and multiple lines that must be supported by an operator would be costly to implement because it may result in a larger number of transceivers and/or more memory in each transceiver. As a result, the POSA would have looked to solutions that could provide the necessary functionality, without increased cost. *Id.* Fadavi-Ardekani provides such a benefit. *Id.* at 3:1–4. Specifically, Fadavi-Ardekani explains that “the memory savings and corresponding reduction in buffer size permitted by the invention reduces integrated circuit area and ADSL transceiver cost,” as well as, in a central office, results in requiring “a lesser number of ADSL transceivers than prior art ADSL communications systems.” *Id.* at 8:8–17. Given the costs savings and reduction in integrated circuit area, the POSA would have been further motivated to select the Fadavi-Ardekani solution.

92. In addition, the POSA would have had a reasonable expectation of success in the combination because Fadavi-Ardekani expressly instructs that its disclosure can be implemented with VDSL, Ex. 1006 at 4:18–21, and because both

VDSL1 and Fadavi-Ardekani disclose transceivers that include Reed-Solomon encoding, an interleaver to interleave Reed-Solomon bytes, a deinterleaver to deinterleave Reed-Solomon bytes, and memory to support the interleaving and deinterleaving functions. *Compare* Ex. 1007 at 59-60, 71-72, *with* Ex. 1006 at Abstract, 6:15-16. For at least these reasons, in reading VDSL1, a POSA thus would have been motivated to look to Fadavi-Ardekani and had a reasonable expectation of success in doing so.

XII. Ground 1: The combination of Mazzoni and VDSL1 renders the challenged claims obvious.

a. Analysis

i. Claim 1: [1.pre] A system that allocates shared memory comprising:

93. This claim element contains the claim term “shared memory,” which I understand was previously construed in a prior litigation as “common memory used by at least two functions, where a portion of the memory can be used by either one of the functions.” In my opinion and in view of this construction, Mazzoni explicitly discloses “a system that allocates shared memory.”

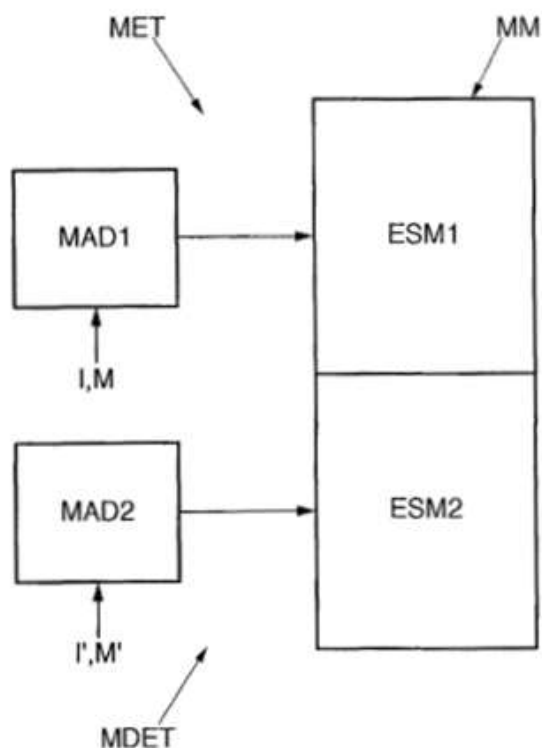
94. Mazzoni is clear in that it discloses a system with a common memory that is used by a first function (e.g., the interleaver) and a second function (e.g., the deinterleaver), where a portion of the memory (MM) can be used by either one of the functions.

95. For example, Mazzoni discloses that the “memory may have a first memory space assigned to the interleaver and a second memory space assigned to the deinterleaver” and that the “size of each of the two memory spaces may be set as a function of the bit rate actually processed by the device.” Ex. 1005 at Abstract.

96. Mazzoni also discloses that “[t]hese and other objects, features, and advantages are provided by a memory means whose size is optimized for a global (send+receive) bit rate, which can be shared between the interleaving means and the deinterleaving means, and whose memory allocation can be reconfigured in accordance with the bit rate actually processed by the send/receive device (modem).” *Id.* at 1:59-65. Indeed, the flexible allocation of the common memory between the interleaver and deinterleaver functions is so core to Mazzoni’s innovation that it is listed as an object of the invention, and the disclosed architecture is adaptable in terms of the memory capacity of the interleaving and deinterleaving means. *Id.* at 1:54–58 (“Still another object of the invention is to provide such an architecture which is adaptable, particularly in terms of the memory capacity of the interleaving and deinterleaving means, to suit a number of different bit rates selected from a predetermined group of bit rates.”).

Mazzoni uses Figure 6, shown below, to illustrate how the common memory, MM, is allocated to the interleaver (MET) and deinterleaver (MDET).

FIG.6



Ex. 1005, Fig. 6.

97. Mazzoni explains that, “[i]n hardware terms, as shown diagrammatically in FIG. 6, the interleaving means and the deinterleaving means include common memory means MM, e.g., a dual-port random access memory.” *Id.* at 5:61–64. “The memory space of the memory MM is then divided into a first memory space ESM1 assigned to the interleaving means MET, and a second memory space ESM2 is assigned to the deinterleaving means MDET.” *Id.* at 5:64–67.

98. Mazzoni further describes the allocation of common memory in its claims. For example, claim 1 recites the following:

1. A device for sending and receiving digital data that is capable of processing different bit rates from a group of predetermined bit rates, the device comprising:

a channel coding/decoding stage comprising

an interleaver,

a deinterleaver,

a shared memory having a minimum size based upon a maximum bit rate of the group of predetermined bit rates and having a first memory space assigned to said interleaver and a second memory space assigned to said deinterleaver, a size of each of the first and second memory spaces being set as a function of the bit rate actually processed by the device,

a Reed-Solomon coder/decoder connected to said interleaver and said deinterleaver and having a length N,

and

said interleaver providing convolutional interleaving of I branches with i-1 blocks of M bytes, and said deinterleaver providing convolutional deinterleaving with I' branches of i'-1 blocks of M' bytes, with I and I' being sub-multiples of N and i and i' being current relative indexes of the branches.

Id. at 8:12–33. Claim 1 discloses a shared memory where the size of the first and second memory spaces are set as a function of the bit rate actually processed by the device. Similarly, claim 2 provides the specific amount of memory set for a particular bit rate:

2. The device according to claim 1 wherein the size of the first memory space is equal to $I \times (I-1) \times M/2$ bytes, the size of the second memory space is equal to $I' \times (I'-1) \times M'/2$ bytes, and the sizes of the first and second memory spaces are set by I, I', M and M'.

99. Moreover, and as a result of the specific VDSL system and in view of the specific disclosures from Mazzoni above, the POSA would have understood that Mazzoni discloses common memory used by at least two functions.

100. As generally understood by a POSA and as described by Mazzoni, a VDSL system is a digital communication system linking an operator and users via very high bit rate transmission lines. Mazzoni further discloses that the invention applies to the send/receive devices at both the operator and user ends of a transmission line. Particularly, Mazzoni explains that “the invention applies to a digital communication system linking an operator and users via very high bit rate transmission lines. Thus, the invention applies more particularly to send/receive devices, usually referred to as modems, at the operator and user ends of a transmission line.” *Id.* at 1:22-27.

101. In a VDSL system, data is transmitted in both directions between two send/receive devices. That is, the modem at the operator end sends data to the user end and receives data from the user end. Similarly, the modem at the user end sends data to the operator end and receives data from the operator end. In VDSL, the data transmission from the operator end to the receiver end is often called the

downstream transmission or *downlink* and the data transmission from the receiver end to the operator end is often called the *upstream transmission* or *uplink*.

102. A driving motivation for the memory sharing described in Mazzoni is that the upstream transmission and the downstream transmission may be configured differently according to a variety of possible “services”, which requires that the interleaver and deinterleaver also be configurable in terms of their memory allocations. Depending on the specific upstream and downstream configurations, the send/receive device disclosed by Mazzoni implements the appropriate interleaver and deinterleaver memory allocation. *See e.g., id.* at 6:11–50. For each distinct selection for the upstream and downstream configurations, Mazzoni discloses dynamically selecting distinct allocations of the common memory between the interleaver and the deinterleaver, as required by the distinct interleaver and deinterleaver behavior required for the selected configurations. *Id.*

103. Mazzoni describes the distinct upstream and downstream configurations, for example, when it describes how VDSL supports a variety of possible selections for upstream and downstream configurations called “services.” *Id.* at 3:62–4:14. Specifically, Mazzoni discloses that those “skilled in the art will appreciate that a VDSL communication system is capable of delivering symmetrical services and asymmetrical services.” *Id.* at 1:28–37. One important aspect of a service is the set of bit rates, including an upstream bit rate and a

downstream bit rate, associated with the service. The bit rates can be the same for both upstream and downstream. This is called a symmetrical service. *See id.* at 1:30–34 (“A service is symmetrical if the bit rate of information exchanged between the operator and the user in both transmission directions (i.e., from the operator to the user and from the user to the operator) is exactly the same.”). The upstream and downstream bit rates can also be different. This is called an asymmetrical service. *See id.* at 1:34–37 (“A service is asymmetrical if the bit rate of information sent in one transmission direction is different from the bit rate of information sent in the other transmission direction.”).

104. As is well known, in a system efficiently supporting both symmetrical and asymmetric services, a portion of the common memory that is shared between two functions could be used by either one of those functions. Mazzoni discloses selecting distinct allocations of the common memory for the interleaver and the deinterleaver for the upstream and downstream configurations, for example, when he discloses that the size of the memory allocation required for an interleaver or deinterleaver depends on the bit rate of the transmission associated with that interleaver. *Id.* at 1:38–42. Specifically, Mazzoni discloses that the “processes of interleaving and deinterleaving data sent and received by a modem necessitates the use of memories. For a modem intended to operate at a predetermined bit rate, the memories must have a capacity that depends on that bit rate.” *Id.*

105. Accordingly, at the operator end, the downstream transmission will require a specific memory allocation for the interleaver depending on the selected configuration and the upstream transmission will require a distinct specific memory allocation for the deinterleaver depending on the selected configuration. Similarly, at the user end, the downstream transmission will require a specific memory allocation for the deinterleaver depending on the selected configuration and the upstream transmission will require a distinct specific memory allocation for the interleaver depending on the selected configuration.

106. Mazzoni explicitly states that the invention supports a number of different configurations of upstream and downstream rates, which include both symmetric and asymmetric rates. *Id.* at 2:19–24 (“It is therefore possible to considerably reduce the size of the memory means required for the interleaving and deinterleaving means implemented within a modem. The modem may be used either at the operator end or at the user end, and it is capable of processing a number of different and symmetrical or asymmetrical bit rates.”). Further, Mazzoni states that the invention may be applied either to a modem at the user end or to a modem at the operator end. *Id.* Accordingly, because Mazzoni discloses an invention that supports both symmetrical and asymmetrical rates, a POSA would have understood that a portion of the common memory that is shared between two functions could be used by either one of those functions.

107. In fact, Mazzoni further explains exactly how the memory may be shared, disclosing that the interleaver means is allocated a first space in the common memory and the deinterleaver means is allocated a second space in the common memory, and that the sizes of the two memory allocations are distinct and vary as a function of the upstream and downstream configurations as reflected, for example, in the send and receive bit rates. Specifically, Mazzoni discloses that the “device according to the invention may include a coding/decoding stage (generally referred to by those skilled in the art as a ‘channel coding/decoding stage’) including interleaving means and deinterleaving means” and that the “interleaving and deinterleaving means include a memory whose minimum size is fixed as a function of the maximum bit rate of the group of predetermined bit rates (e.g., the highest asymmetrical bit rate in the case of a VDSL system).” *Id.* at 2:3–10. It continues by explaining that “[t]he memory also has a first memory space assigned to the interleaving means and a second memory space assigned to the deinterleaving means” and that the “size of each of the two memory spaces is set as a function of the bit rate actually processed by the device.” *Id.* at 10–15. Moreover, “[i]n the context of the present invention, the term ‘bit rate’ as associated with a memory capacity or memory space is a global bit rate, i.e., the sum of the send and receive bit rates.” *Id.* at 15–18.

108. Mazzoni discloses how to compute the specific memory allocation required by a specific configuration. For example, Mazzoni explains that:

More specifically, the channel coding/decoding stage may include Reed-Solomon coding/decoding means of length N (where $N=240$ bytes, for example). The interleaving means are then adapted to effect convolutional interleaving of I branches with $i-1$ blocks of N bytes. The deinterleaving means are adapted to implement convolutional deinterleaving with I branches of $i'-1$ blocks of M bytes. I and I' are sub-multiples of N , and i and i' are the current relative indexes of the branches. The size in bytes of the first memory space is equal to $I \times (I-1) \times M/2$, and the size in bytes of the second memory space is equal to $I' \times (I'-1) \times M'/2$. The sizes of the two memory spaces are set by I , I' , M and M' .

Id. at 2:37–48.

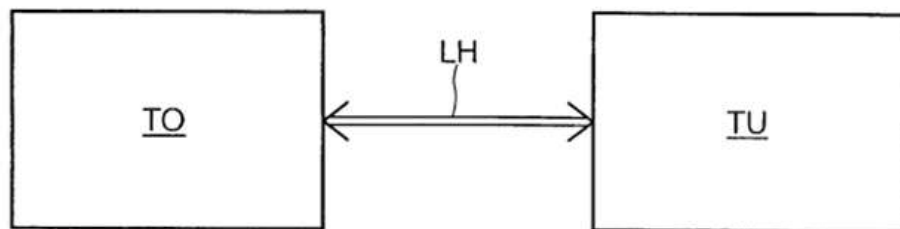
ii. Claim 1: [1.A] a transceiver that is capable of: transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver;

109. This claim element contains the claim term “transceiver,” which I have understood was previously construed in a prior litigation as “communications device capable of transmitting and receiving data wherein the transmitter portion and receiver portion share at least some common circuitry.” In my opinion, the plain and ordinary meaning of this term as known to a POSA would be a “communications device capable of transmitting and receiving data.” Nonetheless, in my opinion and in view of the prior construction, the combination of Mazzoni and VDSL1 explicitly discloses a “a transceiver that performs: transmitting or receiving a message during

initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver.”

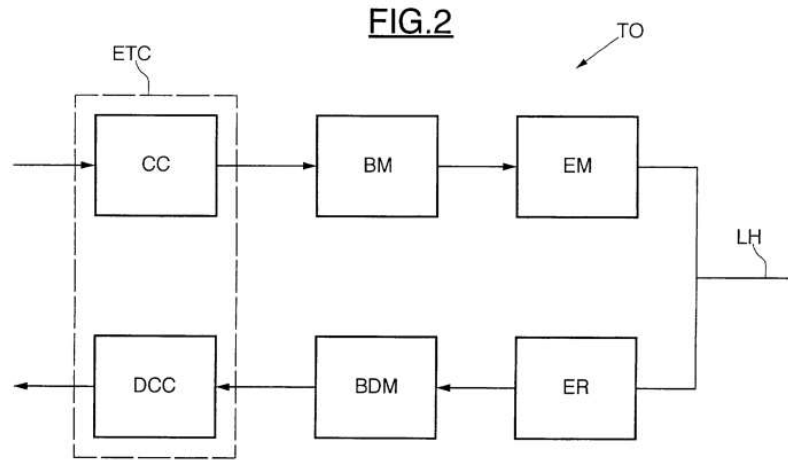
110. Mazzoni discloses a communication device capable of transmitting and receiving data (e.g., transceivers with “transmitters and receivers” for “sending and receiving digital data”). Ex. 1005 at 1:8–15. Specifically, as shown in Figure 1 below, Mazzoni contemplates two send/receive devices (i.e., transceivers), TU and TO, that communicate by “a very high bit rate communication line LH.” *Id.* at 1:60–61.

FIG.1



111. *Id.* at Fig. 1. As is shown in Figure 2, the terminal TO “includes a send system [i.e., transmitter] and a receive system [i.e., receiver] both connected to the transmission line LH.” *Id.* at 4:30–32. Thus, the transmitter portion and receiver portion share at least some common circuitry. Moreover, Mazzoni discloses a shared memory used by both an interleaver and a deinterleaver, where a POSA would have

understood the shared memory of Mazzoni is common circuitry between the “transmitters and receivers.”



112. Mazzoni further discloses a shared memory “MM,” wherein the “size of the memory MM is set by the maximum bit rate that the send/receive device can process.” Ex. 1005 at 6:6–8. The POSA would thus have understood the shared memory of Mazzoni is common circuitry between the “transmitters and receivers.”

113. Although Mazzoni does not expressly disclose that its transceiver performs transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that may be allocated to an interleaver, a POSA would have understood that, to implement the embodiments disclosed in Mazzoni, transmitting and receiving initialization messages setting parameters would have been necessary. In fact, Mazzoni explains that “the parameters I, M, I’, and M’ can be modified, e.g., by software, and are delivered by control means

MCD.” Ex. 1005 at 5:21–24. A POSA would have understood that initialization protocols to agree on parameters M and I used by both the operator and user transceivers would have preceded the delivery of M and I parameters by the MCD. Such a protocol requires both transceivers to understand the limitations of the other transceiver in terms of possible M and I values. Communicating a maximum number of bytes of memory that may be allocated to an interleaver is one of a small number of options available to a POSA to describe such limitations.

114. To the extent it is argued that Mazzoni does not disclose a transceiver that performs: transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver, the VDSL1 standard does so explicitly. The POSA would have been motivated to combine with VDSL1 for explicit instruction on transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver at least because Mazzoni describes a VDSL implementation and the POSA would have relied on the actual technical specification, VDSL1, for underlying detail. *See Motivation to Combine supra.*

115. VDSL1 discloses “a transceiver that performs: transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver.” Specifically, the VDSL1

standard both discloses a transceiver and a message that is transmitted from the VTU-R transceiver and received by the VTU-O transceiver that explicitly specifies a maximum number of bytes of memory that are available to be allocated to an interleaver. *See* Ex. 1007 at 127, 132.

116. First, the VDSL1 standard discloses a transceiver. The standard explicitly refers to its communication devices as transceivers. For example, it defines the acronym “VTU,” which is used throughout the standard, to mean “VDSL Transceiver Unit.” Ex. 1007 at 13. Also defined are the acronyms “VTU-O,” which means the VTU located at the Optical Network Unit (ONU) or operator side, and VTU-R, which means the VTU located at the Remote site, which is the Network Termination (NT) or user side. *Id.*

117. Next, each VTU is capable of transmitting and receiving data. Figure 4 from the VDSL1 technical specification, shown below, illustrates the reference configuration used for the VDSL1 technical specification. The figure shows how VTU-R transceivers, located at the NT, transmit and receive data to and from the VTU-O, which is located at the ONU. Ex. 1007 at 16. Similarly, the VTU-O transceivers transmit and receive data to and from the VTU-R. *Id.*

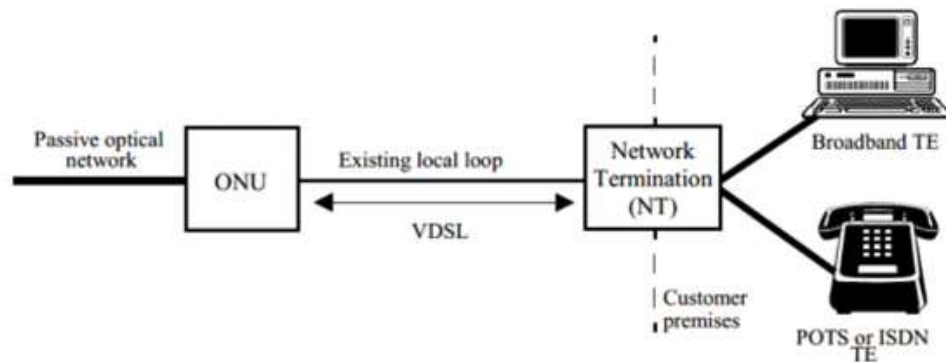


Figure 4: General fibre-to-the-node architecture for VDSL

118. The transmitter and receiver portions of the VTUs share some circuitry. For example, such devices share the circuitry that provides power. Ex. 1007 at 20. Also, the configuration messages that are received by the VTU are used to affect both transmitter and receiver functionality, so the circuitry that utilizes the messages is shared by the transmitter and receiver portions of the VTU. *Id.* at 111. The transmitting and receiving functions commonly share the same clock to control the timing of the digital circuitry. *Id.* As another example, whenever the VTU utilizes shared memory for interleaving and deinterleaving, as taught by Mazzoni, the transmitter portion and receiver portion of the VTU share at least the common circuitry of the common memory used for interleaving and deinterleaving.

119. In addition to disclosing a transceiver, the VDSL1 standard discloses transmitting or receiving a message during initialization specifying a maximum

number of bytes of memory that are available to be allocated to an interleaver.

Specifically, the VDSL1 standard discloses how each of its transceivers transmit or receive a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver in the “channel analysis and exchange state” of the “initialization protocol.” Ex. 1007 at 111.

120. Figure 50, shown below, shows the “Channel analysis and exchange” state as part of initialization. *Id.*

VTU-O		
Activation: Handshake procedures (clause 8.2.3)	Training (clause 8.2.4)	Channel analysis and Exchange (clause 8.2.6)
VTU-R		
Activation: Handshake procedures (clause 8.2.3)	Training (clause 8.2.4)	Channel analysis and Exchange (clause 8.2.6)

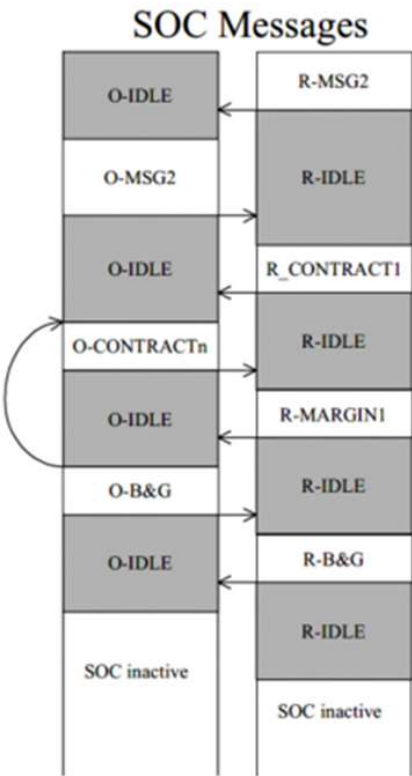
Figure 50: Overview of initialisation

121. A POSA would have understood that the description of Figure 50 discloses that during the channel analysis and exchange state, the VTU-O and VTU-R measure the characteristics of the channel and agree on all the parameters needed to thoroughly define the communication link. Specifically, VDSL1 discloses:

The time line in Figure 50 provides an overview of the initialisation protocol. Following the initial handshake procedure, a full duplex link between the VTU-O and the VTU-R is established. During the training phase, timing advance and upstream power back-off shall be refined. During the channel analysis and exchange state, the two modems measure the characteristics of the channel and agree on a contract that thoroughly defines the communication link.

Id. at 111.

122. Figure 56 provides an overview of the messages exchanged during the channel analysis and exchange state, which are used to agree on all the parameters needed to thoroughly define the communication link.



Id. at 127 (Excerpt from Figure 56 showing messages exchanged during the “Channel analysis and exchange” state).

123. During the “Channel analysis and exchange” state, the VTU-R sends a specific message, R-MSG2, to the VTU-O to describe its capabilities, including its maximum available interleaver memory. *Id.* at 132. For example, the message R-MSG2 includes the field “Maximal interleaver memory,” which specifies the

maximal number of bytes available to be allocated to an interleaver. *Id.* at 132.

The R-MSG2 contains information about the capabilities of the VTU-R for bit allocation. *Id.* The content of R-MSG2 is specified in Table 82, which is shown below.

Table 82: Description of R-MSG2

Field	Field or macro-field type	Remark
Message descriptor	1 octet	See table 49
Maximum constellation size in upstream	1 octet	Maximum number of bits per tone
RS setting supported by VTU-R	1 octet	0x00: only mandatory settings 0xFF: all settings (see note 1)
Interleaver setting supported by VTU-R	1 octet	0x00: only mandatory settings 0xFF: all settings 0xNN: NN = number of settings (0x00 < NN < 0xFF)
Detailed interleaver setting description	0 octets if NN = 0x00 0 octets if NN = 0xFF NN x 4 octets otherwise	See table 74
Maximum power transmitted	1 octet	In units of 0,25 dBm
Maximum interleaver memory	3 octets	In octets (see note 2)
Maximum number of EOC octets per frame in upstream	1 octet	Number of EOC octets per frame
Maximum number of VOC octets per frame in upstream	1 octet	Number of VOC octets per frame
Support of express swapping	1 octet	0x00: Not supported 0xFF: Supported
j_{max}	1 octet	Maximum value of j_{max} supported by the VTU-R (see note 3)
NOTE 1: All settings means the values (for the redundancy) that are specified in clause 6.4.2.		
NOTE 2: The interleaver memory is computed as $M \times l \times (l-1)$.		
NOTE 3: Specification of $j_{max} = k$ means that all values from 0 to k are supported.		

Id.

124. After receiving this message, the VTU-O sends the message O-MSG2 describing its capabilities. *Id.* at 127. After receiving the O-MSG2, the VTU-R sends message R-CONTRACT1 to the VTU-O to propose parameters for the downstream channel Reed-Solomon coding and interleaving, within the capabilities as set forth in the R-MSG2 and O-MSG2 messages. *Id.* at 132. After receiving R-CONTRACT1, the VTU-O responds with its proposed parameters for

both upstream and downstream channels using the O-CONTRACT1 message. *Id.* at 129. The O-CONTRACT1 message bases the downstream contract on the interleaving parameters proposed in R-CONTRACT1. Ideally the downstream contract is the same as the one proposed in R-CONTRACT1. *Id.* at 129. A POSA would have understood that the upstream contract proposed by the VTU-O must comport with the limitations of the VTU-R including maximum interleaver memory as communicated in R-MSG2.

125. When the VTU-R receives the O-CONTRACT1 message, it computes the minimal SNR margin over all tones and sends this information to the VTU-O in the R-MARGIN1 message. *Id.* at 132. If the VTU-O finds the SNR margins in the R-MARGIN1 message acceptable, then it sends the O-B&G message, which ends the negotiation. *Id.* at 129–30. If the VTU-O does not find the SNR margins in the R-MARGIN1 message acceptable, then it sends a new O-CONTRACT2 message and the VTU-R computes the new values for the minimal SNR margin over all tones and sends this information to the VTU-O in the R-MARGIN2 message. *Id.* at 129–32. The VTU-O and VTU-R continue to exchange O-CONTRACT_n and the R-MARGIN_n messages until the VTU-O is satisfied and sends the O-B&G message which ends the negotiation. *Id.*

126. A POSA would have understood that, throughout the exchange, the VTU-O would not propose a contract that exceeds the available interleaver

memory at the VTU-R, which is described in the “maximum interleaver memory” field of the R-MSG2 sent by the VTU-R.

127. Therefore, it is my opinion that a POSA would have understood that the combination of Mazzoni and VDSL1 disclose “a transceiver that performs: transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver.

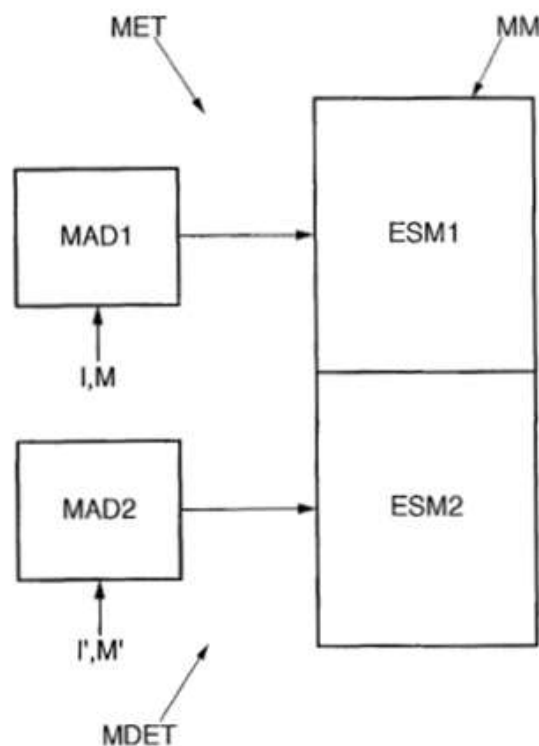
iii. Claim 1: [1.B] determining an amount of memory required by the interleaver to interleave a first plurality of Reed Solomon (RS) coded data bytes within the shared memory;

128. This claim element contains the term “amount of memory,” which I understand was previously construed as the plain meaning without being limiting to a number of bytes. In my view, a POSA would understand the plain and ordinary meaning of the term “an amount of memory,” without further explanation.

129. In my opinion, Mazzoni discloses determining an amount of memory required to interleave a first plurality of Reed Solomon (RS) coded data bytes within a shared memory.

130. As described with respect to Limitation [1.pre], Mazzoni discloses memory shared between the interleaver and the deinterleaver, as well as determining an amount of memory required by the interleaver (MET) in ESM1 and the deinterleaver (MDET) in ESM2. *See Id.*, Fig. 6.

FIG.6



131. Further, Mazzoni discloses interleaving Reed Solomon (RS) coded data bytes within that shared memory, disclosing that “Reed-Solomon coding means are associated with the interleaving means” and “[i]n conjunction with subsequent interleaving, the Reed-Solomon coding can correct bursts of errors introduced by the transmission channel.” *Id.* at 4:38–41. Indeed, and as described below, “Reed-Solomon coding is applied individually to each of the data packets delivered to the input of the coding unit CC” in the send system. *Id.* at 4:42–43.

132. Finally, Mazzoni discloses using the triangular implementation of convolutional interleaving and convolutional deinterleaving, as well as how to determine an amount of memory required by the interleaver in such an implementation. Specifically, Mazzoni discloses that for the triangular implementation, convolutional interleaving and deinterleaving memory allocations can be precisely determined from two parameters. Ex. 1005 at 6:36–44.

133. Parameters I and M are used to determine the precise memory allocation for the triangular implementation of the convolutional interleaving and deinterleaving for the downlink channel (or downstream) of a specified service, and parameters I' and M' are used to determine the precise memory allocation for the triangular implementation of the convolutional interleaving and deinterleaving for the uplink channel (or upstream) of a specified service. *Id.* These parameters are selected so that the precise memory allocation for the interleaver or deinterleaver for that channel for that service is large enough to overcome noise bursts of a specified duration, which impacts the quality of the data. *Id.* at 6:11–37.

134. Specifically, for triangular implementation of convolutional interleaving and deinterleaving that includes Reed-Solomon coding/decoding, Mazzoni discloses:

More specifically, the channel coding/decoding stage may include Reed-Solomon coding/decoding means of length N (where $N=240$)

bytes, for example). The interleaving means are then adapted to effect convolutional interleaving of I branches with $i-1$ blocks of N bytes. The deinterleaving means are adapted to implement convolutional deinterleaving with I' branches of $i'-1$ blocks of M' bytes. I and I' are sub-multiples of N , and i and i' are the current relative indexes of the branches. The size in bytes of the first memory space is equal to $I \times (I-1) \times M/2$, and the size in bytes of the second memory space is equal to $I' \times (I'-1) \times M'/2$. The sizes of the two memory spaces are set by I , I' , M and M' .

Using convolutional triangular interleaving (and, consequently, convolutional triangular deinterleaving) instead of other conventional types of interleaving is particularly beneficial because it reduces latency generated by the memory. Convolutional triangular interleaving requires a much smaller memory, which reduces latency. Latency is a primordial and decisive criterion for any VDSL communication system.

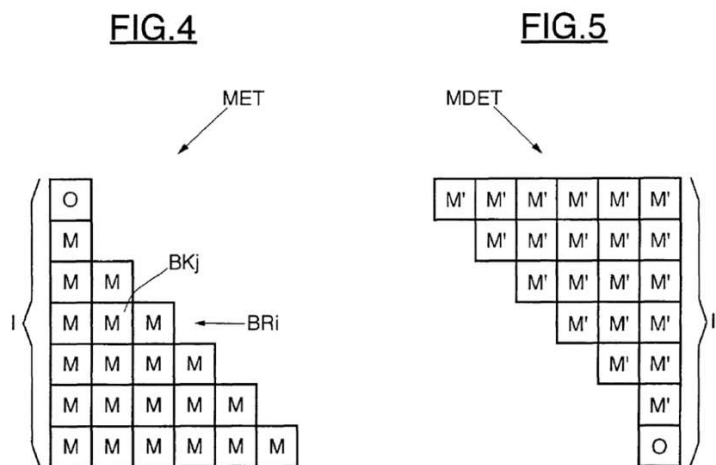
Id. at 2:37–56.

135. Mazzoni further teaches that the parameters I , M , I' , and M' are not fixed for all time, but rather can be modified by, for example, software. *Id.* at 5:21–30. As a result, for varying parameters, Mazzoni discloses determining an amount of memory required by the interleaver to interleave. Specifically, Mazzoni explains:

As explained in more detail hereinafter, the parameters I , M , I' and M' can be modified, e.g., by software, and are delivered by control means MCD (see FIG. 3). The control means MCD may also be implemented in software. These parameters define the sizes of the respective memory spaces assigned to the interleaving means and to the deinterleaving means. This is done according to the bit rate of the information sent by the terminal TO (parameters I and M) and the bit rate of the information received by the terminal TO (parameters I' and M').

Id.

136. Mazzone uses Figures 4 and 5 of the specification, shown below, to describe the triangular implementation of convolutional interleaving.



137. Figure 4 shows the triangular implementation of an interleaver for the downstream with parameters I and M . Figure 5 shows the triangular implementation of a deinterleaver for the upstream with parameters I' and M' . The interleaver and deinterleaver structures each include I rows corresponding to delay lines or shift registers (or virtual shift registers (*See* VDSL1 at 152-153) implemented, for example, in a random access memory). Each box containing the letter M/M' represents M/M' memory elements in a shift register used to delay bytes written into the interleaver. I bytes at a time are written into the interleaver at the left side, one to each shift register. I bytes are read from the interleaver at the right side, one from each shift register.

138. In the interleaver, bytes written to the top shift register experience a delay of zero bytes. Bytes written to the second shift register experience a delay of M bytes. Bytes written to the third shift register experience a delay of $2M$ bytes. This pattern continues until the bottom row, which is delayed by $(I-1) \times M$ bytes. To undo the interleaving, the deinterleaver matched to this interleaver in the downstream will delay inputs to the top shift register by $(I-1) \times M$ bytes, inputs to the next shift register by $(I-2) \times M$ bytes, and so on. In the deinterleaver in the downstream, the bottom shift register implements a delay of zero bytes.

139. In this way, the overall effect of the combination of the interleaver and deinterleaver is to delay each column by $(I-1) \times M$ bytes. However, each of the I bytes in a column is transmitted separately so that the overall delay associated with the interleaver is $I \times (I-1) \times M$ bytes of delay. Thus, the deinterleaver returns the interleaved sequence back to its original order, but the combination of interleaving and deinterleaving delays the sequence by $I \times (I-1) \times M$ bytes.

140. Figure 4 further shows how the parameters I and M can precisely determine the amount of memory that needs to be allocated for the triangular implementation of a convolutional interleaver. *Id.*, Fig. 4. Each box containing the letter M represents M memory elements used to delay bytes written into the interleaver. Counting up the boxes, there are $I-1$ in the first column, $I-2$ in the second column, and so on until the last column, which has only one box containing

M memory elements. Since the sum of $(I - 1) + (I - 2) + (I - 3) + \dots + 1 = \frac{I \times (I - 1)}{2}$, the total amount of memory required to be allocated for the interleaver is precisely $I \times (I - 1) \times M / 2$. The same argument shows that the associated deinterleaver with parameters I and M also requires an allocation of $I \times (I - 1) \times M / 2$ bytes of memory for the specific triangular implementation. Although there are other implementations of convolutional interleaving that use different amounts of memory, Mazzoni teaches that the triangular implementation should be used. *Id.* at 2:49–52. VDSL1 likewise teaches the triangular implementation of convolutional interleaving and deinterleaving.

141. Figure 5 of Mazzoni shows the triangular implementation of a convolutional deinterleaver with parameters I' and M' in the upstream. *Id.*, Fig. 5. For the triangular implementation of a convolutional deinterleaver with parameters I' and M', the above argument shows that both for the deinterleaver and for its associated interleaver, precisely $I' \times (I' - 1) \times M' / 2$ bytes of memory are required to be allocated.

142. Mazzoni describes Figures 4 and 5 of the specification in the excerpts recited below:

As shown diagrammatically in FIGS. 4 and 5, the interleaving and deinterleaving are convolutional triangular interleaving and deinterleaving. There are I branches of i-1 blocks of M bytes for interleaving and I' branches of i'-1 blocks of M' bytes for deinterleaving.

Id. at 5:15–20.

The interleaving means therefore include I parallel branches BR_i (numbered from 0 to $I-1$, for example) which are implemented with a delay increment of M per branch (M represents the maximum number of bytes of a block BK_j with index j). Each branch can be considered as a delay line, the length of the branch with index i (where i varies in the range from 0 to $I-1$) being equal to $i \times M$ bytes. In FIG. 4, by way of example, $I=7$.

Id. at 5:31–38.

Accordingly, the first block of M bytes (having the index 0, for example) is not interleaved and is delivered unmodified to the output of the interleaving means. The next block of M bytes (index 1) is delivered to the input of the branch BR_1 , and so on up to the seventh block of M bytes (index 6), which is delivered to the branch BR_6 . The cycle then begins again with the blocks of bytes with indices from 7 to 13. The preceding blocks of bytes are either delivered to the output of the interleaving means or moved forward by one block BK_j in the branch concerned.

Id. at 5:39–48.

The deinterleaving means associated with the interleaving means MET, and which are consequently incorporated into the user terminal TU, have a structure analogous to that which has just been described for the interleaving means. Yet, the indices of the branches are reversed so that the longest interleaving time-delay corresponds to the shortest deinterleaving time-delay. The deinterleaving means MDET incorporated in the operator terminal TO have I' branches, the branch with index i' having a length equal to $i' \times M'$ bytes.

Id. at 5:49–57.

For simplicity, the situation in which $I'=I$ is shown in FIG. 5. However, if the service is an asymmetrical service, I and I' are generally different, of course, and likewise M and M' . In hardware terms, as shown diagrammatically in FIG. 6, the interleaving means and the deinterleaving means include common memory means MM, e.g., a dual-port random access memory. The memory space of the memory MM is then divided into a first memory space ES M1

assigned to the interleaving means MET, and a second memory space ESM2 is assigned to the deinterleaving means MDET.

Id. at 5:58–67.

143. Mazzoni further discloses an example of determining an amount of memory required by an interleaver or deinterleaver to interleave or deinterleave a first plurality of Reed Solomon coded data bytes based on a first data rate. *See id.* at 6:11–50.

144. Specifically, Mazzoni first discloses an example system with six asymmetric services A1-A6 and six symmetric services S1-S6, each with a different bit rate that may be implemented. *See id.* at 3:62–4:14. For the example of the asymmetrical service A6, the downlink bit rate is 832×64 kbit/s (or 53,248,000 bits per second), and the uplink bit rate is 32×64 kbit/s (or 2,048,000 bits per second). *Id.* at 4:7–14. For the symmetric service S6, both uplink and downlink have bit rates of 362×64 kbits/s (or 23,168,000 bits /s). *Id.* at 3:66–4:2. The bit rates for the various services Mazzoni considers in his example system are described in the following excerpt:

The VDSL communication system enables the operator to provide symmetrical services, typically six symmetrical services S1-S6. That is, the information bit rates in the two transmission directions (i.e., from the operator to the user and from the user to the operator) are exactly the same. The service S1 with the lowest bit rate has a bit rate of 32×64 The VDSL communication system enables the operator to provide symmetrical services, typically six symmetrical services S1-S6. That is, the information bit rates in the two transmission directions (i.e., from the operator to the user 65 and from the user to the

operator) are exactly the same. The service S1 with the lowest bit rate has a bit rate of 32×64 kbit/s.

Id. at 3:62–4:2.

With the VDSL system, the operator can also provide asymmetrical services A1-A6. These are services with different information bit rates in the user to operator direction (uplink direction) and in the operator to user direction (downlink direction). The first asymmetrical service A1 has a bit rate in the uplink direction of 32×64 kbit/s, for example, and a bit rate in the downlink direction of 100×64 kbit/s. The asymmetrical service having the highest global information bit rate (uplink bit rate+downlink bit rate) is the service A6. The bit rate of the service A6 in the uplink direction is equal to 32×64 kbit/s and in the downlink direction is equal to 832×64 kbit/s.

Id. at 4:3–14.

145. For the example computation, Mazzoni selects service A6, and teaches selecting an interleaver/deinterleaver pair large enough to spread the errors caused by a 0.25 millisecond burst of impulsive noise over enough Reed Solomon codewords that the errors can all be corrected. *Id.* at 5:12–18. Mazzoni also considers a Reed Solomon code that communicates 224 bytes of information using a 240-byte codeword, i.e. $N=240$. *Id.* Thus, there are sixteen “extra” bytes of redundancy.

146. As would have been known by a POSA, a Reed Solomon code can correct one errored byte for every two bytes of redundancy. Thus, with sixteen bytes of redundancy, it is possible to correct eight errored bytes in a Reed-Solomon codeword of 240 bytes. Mazzoni describes the 0.25 ms noise burst and the 8-byte correction capability of a (240,224) RS code as follows:

In this example, the maximum bit rate is that of the largest asymmetrical service A6. An example of the capacity of the memory MM and of the values chosen for the parameters I, M, I' and M' for an asymmetrical service A6 and an RS (240, 224) Reed-Solomon code with a correction power of 8 bytes/word may be as follows when the transmission lines are disturbed by an impulsive noise with a duration of 0.25 ms.

Id.

147. As taught by Mazzoni and as understood by a POSA, the number of errored bytes caused by a 0.25 ms noise burst with service A1 (downstream 832x64 kbit/s and upstream 32x64 kbit/s) depends on the bit rate of the channel. *See id.* at 6:9–36. To determine the number of Reed-Solomon coded data bytes, Mazzoni calculates the number of bits affected in the downstream by this 0.25 ms impulsive noise by multiplying the transmitted bit rate (832x64 kbits/s) by the duration of the noise burst (0.25ms), resulting in 13,312 affected bits. *Id.* at 6:19–23. Next, Mazzoni determines the number of errored bytes by dividing the total number of affected bits by 8, the number of bits in a byte, resulting in 1,664 bytes.

Id.

148. Mazzoni then computes the number of Reed-Solomon (RS) codewords (nrs) over which the errored bytes must be spread for all of the bytes to be corrected as the number of errored bytes divided by the number of bytes that can be corrected by one RS codeword, which, in the example disclosed in Mazzoni, is eight. *Id.* at 6:23–26. Thus, Mazzoni explains that the number of

Reed-Solomon codewords needed to correct the errored bytes is the 1,664 errored bytes divided by 8, or 208 codewords. Ex. 1005 at 6:23–26.

149. Finally, Mazzoni discloses computing the size, in bytes, of the interleaver or deinterleaver memory required to hold the necessary number of RS codewords. *Id.* at 6:26–30. Since the interleaver and deinterleaver each hold half of the bytes, the amount of memory required for each is simply $N \times nrs / 2$, where N is the size of the Reed-Solomon code (here 240),” and nrs is the number of RS codewords. Ex. 1005 at 6:23–30. Accordingly, the resulting memory space size in the downstream is equal to 24,960 bytes. *Id.*

150. For example, Mazzoni provides an example for the case of the downlink channel for the A6 asymmetrical service as follows:

In the downlink direction, the maximum bit rate is equal to 832×64 kbit/s. The number of bits affected by noise is consequently equal to the product of that bit rate by the duration of the impulsive noise, which yields a number of bits affected by noise equal to 13,312 (1,664 bytes). Given the correcting power of the Reed-Solomon code (here 8), the number nrs of Reed-Solomon words needed to correct 1,664 bytes subject to noise is equal to $1,664 / 8$, i.e. 208. The size of the memory space needed to store this maximum bit rate is then equal to $N \times nrs / 2$, where N is the size of the Reed-Solomon code (here 240). The resulting memory space size is therefore equal to 24,960 bytes.

Id. at 6:19–30.

151. Applying the same methodology in the upstream, where the bit rate is 32×64 , the number of bytes affected by the 0.25 ms noise duration is 64 bytes (32×64 kbits is equal to 2,048,000 bits, multiplied by 0.25 ms is 512 bits affected, and

divided by 8 bits per byte is 64 bytes). Ex. 1005 at 6:31–35. The number of Reed-Solomon codewords needed to correct the errored bytes is the 64 errored bytes divided by 8, or 8 Reed-Solomon codewords. *Id.* Calculating the memory space size based on $N_{xrs}/2$, it can be determined that the minimum required upstream memory is 960 bytes, although Mazzoni erred and calculated it to be 1,920 bytes. For ease of reference, assuming Mazzoni’s calculation is correct, and the minimum required upstream memory is 1,920 bytes, adding the required memory from the downstream and the upstream, Mazzoni determines that the “minimum size of the memory MM is therefore 26,880 bytes.” Ex. 1005 at 6:29–30.

152. Finally, to determine the minimum amount of memory required by the interleaver to interleave a first plurality of Reed Solomon (RS) coded data bytes, Mazzoni explains that it is necessary to determine the parameters I , I' , M , and M' . Ex. 1005 at 6:37–38. Mazzoni states that “parameters I , M , I' and M' . . . define the sizes of the respective memory spaces assigned to the interleaving means and to the deinterleaving means.” Ex. 1005 at 5:21–27. Mazzoni explains that “the size of the first memory space [ESM1]...is equal to $I \times (I-1) \times M/2$.” *Id.* at 6:38–41. Likewise, “the size of the second memory space ESM2...is equal to $I' \times (I'-1) \times M'/2$.” *Id.* at 6:41–44. Mazzoni therefore explains that because $I \times (I-1) \times M/2$ must be equal to the calculated downlink memory of 24,960 bytes, “it is possible to choose [parameters] $I=40$ and $M=32$.” *Id.* at 6:45–48. Likewise,

because $I' \times (I' - 1) \times M' / 2$ must be at least equal to the calculated upstream memory of 1,920 bytes, “it is possible to choose [parameters] $I'=24$ and $M'=7$.” *Id.*

Because the required memory when $I'=24$ and $M'=7$ is 1,932 bytes, Mazzoni explains that the common memory required (MM) is actually 26,892 bytes, with the ESM1 memory being equal to 24,960 bytes and the ESM2 memory being equal to 1,932 bytes. Thus, Mazzoni discloses determining an amount of memory (or a number of bytes) required by the interleaver (e.g., 24,960 bytes) to interleave a first plurality of Reed Solomon (RS) coded data bytes (e.g., 1,664 errored bytes or 208 code words) within a shared memory (e.g., MM).

153. Using the computational process taught by Mazzoni, I have computed the interleaver or deinterleaver memory burst for the A6 downlink (replicating the example of Mazzoni) as well as the A6 uplink and the symmetric uplink and downlink of the S6 service.

Channel	A6 downlink	A6 uplink	S6 uplink or downlink
Bit Rate in bits per second (bit rate)	53,248,000	2,048,000	23,168,000
Noise Burst Duration in seconds (duration)	0.00025	0.00025	0.00025
Number of bytes affected by the noise: errored bytes = bit rate x duration / 8	1664	64	724
Number of bytes corrected by Reed Solomon codeword (correction capability)	8	8	8

Number of RS codewords (nrs) required to spread the burst nrs = errored bytes / correction capability	208	8	91
Number of bytes in Reed-Solomon Codeword (N)	240	240	240
Number of bytes required for interleaver or deinterleaver memory: memory = N x nrs / 2	24960	960	10920

Table 1: Computation of interleaver memory required to overcome a 0.25 ms noise burst with a (240,224) RS code that can correct eight errored bytes.

154. Mazzoni performs the computations for the A6 uplink in the excerpt shown below. The computations of Mazzoni match mine in reaching the conclusion that nrs=8. However, Mazzoni computes the memory space to be 1,920, whereas I have computed it to be 960. Mazzoni neglected to divide $N \times nrs$ by 2 in this example.

The bit rate in the uplink direction is equal to 32x64 kbit/s. A similar calculation shows that the number of bits affected by noise is equal to 512, and that nrs=8. The memory size to be provided for the uplink direction is therefore equal to 1,920 bytes. The minimum size of the memory MM is therefore 26,880 bytes.

Id. at 6:31–36.

155. Based on the calculations in Table 1, *see* ¶151, a portion of the common memory shared by interleaving and deinterleaving can be used by either one of the functions. In the example (based on service A6) disclosed by Mazzoni, Mazzoni computes the final size of the common memory MM to be 26,892 bytes. Ex. 1005 at 6:45–50. For the asymmetric service A6, Mazzoni computes an

allocation of 24,960 bytes for the downstream interleaver and an allocation of 1,932 bytes for the upstream deinterleaver. *Id.* at 6:19–36. Mazzoni further teaches choosing the size of the common memory to be exactly large enough to support the sum of the interleaver and deinterleaver allocations required by the most memory intensive service.

156. The interleaver and deinterleaver memory allocations described by Mazzoni for the A6 service are illustrated below in Figure 1. For purposes of illustration, Mazzoni’s memory sizes have been rounded to the nearest kilobyte. Each square in the figure represents one kilobyte of memory.

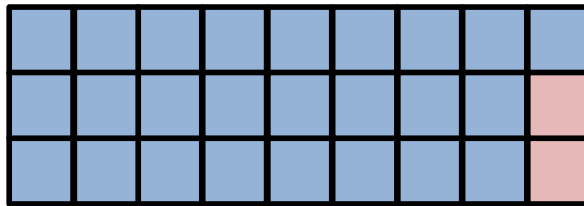


Figure 1: Memory allocation to interleaver (blue squares) and deinterleaver (red squares) for the asymmetric service A6

157. It is apparent that a portion of the common memory shared by interleaving and deinterleaving can be used by either one of the functions when you consider symmetric service S6 contemplated by Mazzoni. S6 uses the same bit rate of 23,168 kbits/s (362x64 kbit/s) for both the uplink and the downlink. *See id.* at 4:1–2. As shown in Table 1, ¶ 151, to overcome a 0.25 ms noise burst on a channel with a bit rate of 23,168 kbits/s requires 724 errored bytes to be corrected,

which means that the burst must be spread over 91 RS codewords of length 240 bytes. This requires at least 10,920 bytes of memory for the triangular implementation of a convolutional interleaver or deinterleaver.

158. Figure 2 shows the resulting allocations of the common memory to the interleaver and deinterleaver for the S6 service. Comparing Figure 1 and Figure 2, we see that a portion of the memory can be used for both interleaving and deinterleaving. Specifically, 8988 bytes of memory allocated to the deinterleaver for the S6 service must be bytes of memory that were also used for the interleaver of the A6 service. That is, the deinterleaver memory allocated for the S6 service beyond the 1932 bytes of deinterleaver memory that was allocated to the deinterleaver for the A6 service must have been memory that was used for the interleaver of the A6 service.

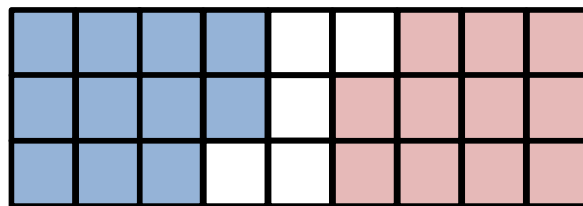


Figure 2: Memory allocation to interleaver (blue squares) and deinterleaver (red squares) for the symmetric service S6

159. Thus, Mazzoni discloses a system where a portion of the common memory shared by interleaving and deinterleaving can be used by either one of the functions.

160. Note that in the example above I used the value of the uplink deinterleaver memory allocation that Mazzoni computed using $I'=24$ and $M'=7$, namely 1932 bytes. Mazzoni arrived at this deinterleaver based on the erroneous calculation that the uplink needed at least 1920 bytes of deinterleaver memory to overcome a 0.25 ms noise burst, when actually only 960 bytes are needed. Regardless of whether the deinterleaver is designed to exceed 1920 bytes or 960 bytes, the example shows that a portion of the common memory shared by interleaving and deinterleaving can be used by either one of the functions.

161. As pointed out by Mazzoni, the process for determination of the memory requirements for the interleaver and deinterleaver are essentially the same for the operator unit and the remote unit, except that the interleaver and deinterleaver roles are swapped for the upstream and downstream links. *See* Ex. 1005 at 8:3-8 (“Of course, everything just described here for the terminal TO applies to the terminal TU with deinterleaving means with I branches and interleaving means with I branches. For the user terminal TU it is then necessary to substitute I' for I, and vice versa, and M for M, and vice versa, in the foregoing description.”).

162. Following the methodologies as disclosed by Mazzoni, and as described *infra* in Ground 2 Limitation [1.B], VDSL1 discloses determining an

amount of memory (or number of bytes) required by the interleaver to interleave a first plurality of Reed Solomon (RS) coded data bytes.

iv. Claim 1: [1.C] allocating a first number of bytes of the shared memory to the interleaver to interleave the first plurality of Reed Solomon (RS) coded data bytes for transmission at a first data rate, wherein the allocated memory for the interleaver does not exceed the maximum number of bytes specified in the message;

163. In my opinion, Mazzoni in combination with VDSL1 discloses allocating a first number of bytes of the shared memory to the interleaver to interleave the first plurality of Reed Solomon (RS) coded data bytes for transmission at a first data rate, wherein the allocated memory for the interleaver does not exceed the maximum number of bytes specified in the message. Mazzoni in combination with VDSL1 discloses this claim element for the same reasons as discussed in elements [1.A] and [1.B]. *See supra* ¶¶ 108–161. Specifically, Mazzoni discloses allocating a first number of bytes of the shared memory to the interleaver (in Mazzoni, as described *supra*, 24,690 bytes to ESM1 of shared memory MM) to interleave the first plurality of Reed Solomon (RS) coded data bytes (in Mazzoni, at least 208 RS codewords as described *supra*) for transmission at a first data rate (in Mazzoni, as described *supra*, 832x64 kbit/s in the downlink direction).

164. In addition to *determining* an amount of memory required to interleave a first plurality of Reed-Solomon coded data bytes as described *supra*, Mazzoni discloses *allocating* the required amount to the interleaver.

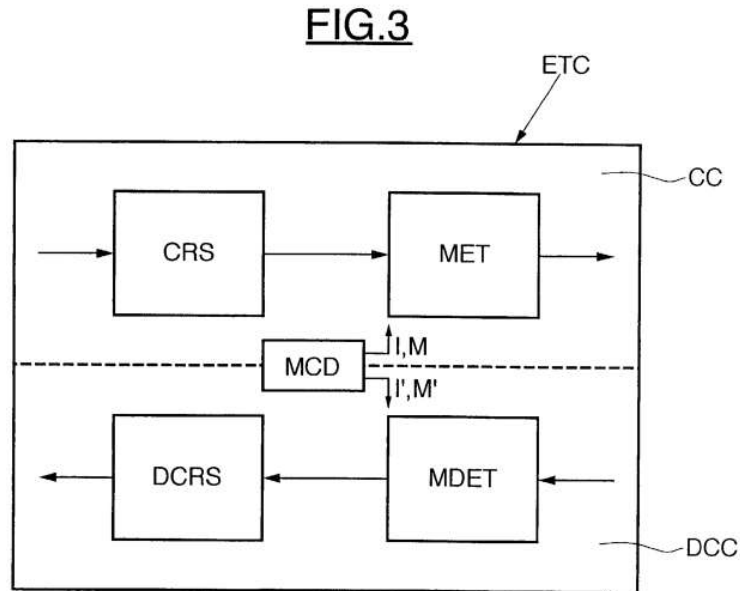
165. Specifically, Mazzoni discloses using the triangular implementation of convolutional interleaving and convolutional deinterleaving to precisely determine allocations from parameters I and M. Ex. 1005 at 6:36–44. Parameters I and M are used to determine the precise memory allocation for the triangular implementation of the convolutional interleaving and deinterleaving for the downstream of a specified service, and parameters I' and M' are used to determine the precise memory allocation for the triangular implementation of the convolutional interleaving and deinterleaving for the upstream of a specified service. *Id.* Memory is thus allocated to the convolutional interleaver and deinterleaver based on said determination.

166. Next, Mazzoni discloses that the memory spaces defined by these parameters are *assigned* to the interleaving means and to the deinterleaving means. *Id.* at 5:21–30.

167. For example, Mazzoni explains that after the values of I, M, I', and M' are determined—which are the values that “define the sizes of the respective memory spaces assigned to the interleaving means and to the deinterleaving means,” Ex. 1005 at 5:24–27—they are “delivered by control means MCD,” *id.* at

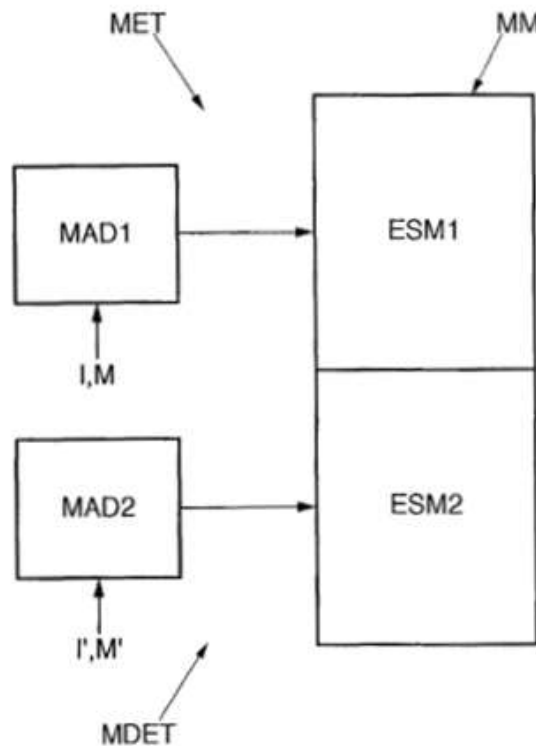
5:21–24. As depicted in Figure 3, MCD delivers parameters I, M, I', and M' to “the interleaving means MET” and “deinterleaving means MDET.” *Id.* at 5:64–67;

Fig. 3.



168. As depicted in Figure 6, “memory space ESM1 [is] assigned to the interleaving means MET,” and “memory space ESM2 [is] assigned to the deinterleaving means MDET.” *Id.* at 5:64–67; Fig. 6.

FIG.6



169. Thus, after teaching *determining* an amount of memory required by the interleaver or deinterleaver, Mazzoni teaches *allocating* that memory to the interleaver and deinterleaver. Specifically, Mazzoni explains:

As explained in more detail hereinafter, the parameters I, M, I' and M' can be modified, e.g., by software, and are delivered by control means MCD (see FIG. 3). The control means MCD may also be implemented in software. These parameters define the sizes of the respective memory spaces assigned to the interleaving means and to the deinterleaving means. This is done according to the bit rate of the information sent by the terminal TO (parameters I and M) and the bit rate of the information received by the terminal TO (parameters I' and M').

Id. at 5:21–30.

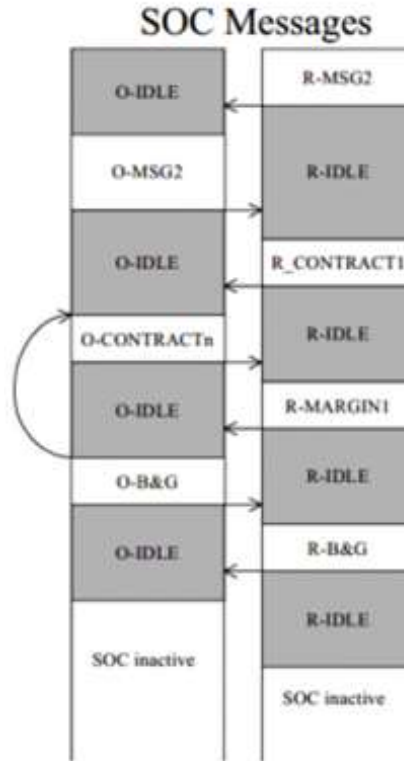
170. As to wherein the allocated memory for the interleaver does not exceed the maximum number of bytes specified in the message, and as discussed in Limitations [1.A] and [1.B], a POSA would have understood that, to implement the embodiments disclosed in Mazzoni, transmitting and receiving initialization messages setting parameters would have been necessary. A POSA would have known that the transceiver cannot implement an interleaver or deinterleaver that requires more memory than the maximum memory that it has available. A POSA also would have known that it cannot implement an interleaver or deinterleaver that would exceed the capabilities of the transceiver on the other end. In the context of a system that is adaptable to support a wide variety of interleaving parameters, successfully establishing a communication link requires that the transceiver knows its own limitations on interleaver/deinterleaver memory and also that it knows the limitations of the transceiver on the other end.

171. It is thus inherent that the transceiver in Mazzoni, in complying with maximum interleaver memory requirements, would allocate memory for the interleaver or deinterleaver that does not exceed the maximum number of bytes allowed both for itself and for the transceiver at the other end.

172. To the extent it is determined that Mazzoni does not disclose the requirement that the allocated memory for the interleaver does not exceed the

maximum number of bytes specified in the message, such a limitation was disclosed by VDSL1. VDSL1 discloses a transceiver and a message that is transmitted from the VTU-R transceiver and received by the VTU-O transceiver that explicitly specifies a maximum number of bytes of memory that are available to be allocated to an (de)interleaver at initialization. See Ex. 1007 at 132; see also [1.A] and [1.B], *supra*.

173. As explained *supra*, VDSL1 discloses that in the “initialization protocol,” there is a “channel analysis and exchange state” whereby the modems “agree on a contract that thoroughly defines the communication link.” Ex. 1007 at 111. As depicted in excerpted Figure 56, the “channel analysis and exchange state” includes a number of messages delivered between the VTU-O and VTU-R.



Ex. 1007 at 127.

174. As explained *supra*, the VTU-R and VTU-O begin a series of negotiations to set the parameters for downstream and upstream interleaving and deinterleaving. Specifically, VDSL1 explains that the “VTU-R sends the R-MSG2 message to transfer information about its bit allocation capabilities.” Ex. 1007 at 127. After receiving R-MSG2, the VTU-O sends the message O-MSG2 describing its capabilities. Ex. 1007 at 128. After receiving the O-MSG2, the VTU-R sends message R-CONTRACT1 to the VTU-O to propose parameters for the downstream channel Reed-Solomon coding and interleaving, within the

capabilities as set forth in the R-MSG2 and O-MSG2 messages. *Id.* at 131. After receiving R-CONTRACT1, the VTU-O responds with its proposed parameters for both upstream and downstream channels using the O-CONTRACTn message. *Id.* at 129.

175. As depicted in Table 75, the O-CONTRACTn, delivers a contract descriptor message for the downstream contract and upstream contract. *Id.* at 129. This message coming from the VTU-O includes bits specifying the values of M and I as the interleaver or deinterleaver settings. *Id.*

Table 75: Description of O-CONTRACTn

Field	Field or macro-field type
Message descriptor	Message code (1 octet)
Downstream contract	Contract Descriptor
Upstream contract	Contract Descriptor
EOC capacity	Number of EOC octets per frame (1 octet)
VOC capacity	Number of VOC octets per frame (1 octet)

176. As depicted in Table 76, the contract descriptor message for the downstream contract includes the parameters I and M for the interleaver (here, labeled “Interleaver setting”). Ex. 1007 at 129.

Table 76: Contract Descriptor

Field	Field or macro-field type	Remark
Rate in fast channel	2 octets	In multiple of 64 kbps
RS setting in fast channel	2 octets	b15 → b8: RS overhead b7 → b0: RS codeword length
Rate in slow channel	2 octets	In multiple of 64 kbps
RS setting in slow channel	2 octets	b15 → b8: RS overhead b7 → b0: RS codeword length
Interleaver setting	2 octets	b15 → b8: M (see note 1) b7 → b0: I
NOTE: The value I must be a divider of the RS codeword length.		

177. A POSA would have understood that the message delivered by the final O-CONTRACT_n message, which included the I and M parameters from which the memory will be determined according to the equation specified in VDSL1, is the allocation of that memory. Indeed, having set the parameters I and M during initialization, VDSL1 discloses, in Table 19, that the interleaving and deinterleaving memory size may be determined, and subsequently allocated, based on the parameters I and M using equation $M \times I \times (I-1)/2$. Ex. 1007 at 60.

Table 19: Characteristics of triangular, convolutional interleaver

Parameter	Value
Interleaver block length	I octets (I must divide N)
Interleaving depth <i>D</i>	$M \times I + 1$ octets
(De)interleaver memory size	$M \times I \times (I-1)/2$ octets
End-to-end delay	$M \times I \times (I-1)$ octets
Correction capability	$\lfloor t/q \rfloor \times M \times (I+1)$ octets

178. The O-CONTRACT₁ message bases the downstream contract on the interleaving parameters proposed in R-CONTRACT₁. Ideally the downstream contract is the same as the one proposed in R-CONTRACT₁. *Id.* at 129. A POSA would have understood that the upstream contract proposed by the VTU-O must comport with the limitations of the VTU-R including maximum interleaver memory as communicated in R-MSG2. As such, a POSA would have understood that the O-CONTRACT_n message confirms that the allocated memory for the interleaver does not exceed the maximum number of bytes specified in the R-MSG2 message. For example, the memory allocation for the upstream interleaver

in the VTU-R does not exceed the maximum number of bytes specified in the R-MSG2 sent by the VTU-R.

179. The combination of Mazzoni and VDSL1 thus also discloses wherein the allocated memory for the interleaver (in VDSL1, set by message O-CONTRACT_n) does not exceed the maximum number of bytes specified in the message (in VDSL1, set by message R-MSG2).

v. Claim 1: [1.D] allocating a second number of bytes of the shared memory to a deinterleaver to deinterleave a second plurality of RS coded data bytes received at a second data rate; and

180. Mazzoni discloses this limitation for the same reasons as discussed in Limitation [1.C]. See *supra*, ¶¶ 162–168. Particularly, Mazzoni discloses a “control means MCD,” Ex. 1005 at 5:21–24, that delivers parameters I, M, I’, and M’ to both “the interleaving means MET” and “deinterleaving means MDET,” *id.* at 5:64–67; Fig. 3. Also, “memory space ESM1 [is] assigned to the interleaving means MET,” and “memory space ESM2 [is] assigned to the deinterleaving means MDET.” *Id.* at 5:64–67; Fig. 6.

181. Moreover, Mazzoni discloses that there may be multiple data rates. As explained *supra*, in Limitation [1.B], ¶¶ 127–161, Mazzoni considers asymmetrical service A6, where the downlink bit rate is 832×64 kbit/s (or

53,248,000 bits per second), and the uplink bit rate is 32×64 kbit/s (or 2,048,000 bits per second). *Id.* at 4:7–14.

182. Mazzoni thus discloses allocating a second number of bytes of the shared memory to a deinterleaver (in Mazzoni, as described *supra*, 1,932 bytes to ESM2 of shared memory MM) to deinterleave a second plurality of RS coded data bytes (in Mazzoni, as described *supra*, 64 errored bytes or 8 code words) received at a second data rate (in Mazzoni, as described *supra*, 32×64 kbit/s in the uplink direction).

vi. Claim 1: [1.E] interleaving the first plurality of RS coded data bytes within the shared memory allocated to the interleaver and deinterleaving the second plurality of RS coded data bytes within the shared memory allocated to the deinterleaver, wherein the shared memory allocated to the interleaver is used at the same time as the shared memory allocated to the deinterleaver.

183. As is discussed in Limitations [1.B]–[1.D], ¶¶ 127–181, Mazzoni discloses interleaving the first plurality of RS coded data bytes (as described *supra*, at least 208 RS code words) within the shared memory allocated to the interleaver (as described *supra*, 24,690 bytes to ESM1 of shared memory MM) and deinterleaving the second plurality of RS coded data bytes (as described *supra*, at least 8 RS code words) within the shared memory allocated to the deinterleaver (as described *supra*, 1,932 bytes to ESM2 of shared memory MM).

184. Specifically, Mazzoni discloses that, for triangular implementation of convolutional interleaving and deinterleaving, “the channel coding/decoding stage may include Reed-Solomon coding/decoding means of length N (where $N=240$ bytes, for example).” Ex. 1005 at 2:37–39. It then explains that an interleaver is “adapted to effect convolutional interleaving of I branches with $i-1$ blocks of N bytes.” *Id.* at 2:39–41. Likewise, it discloses that a deinterleaver is “adapted to implement convolutional deinterleaving with I' branches of $i'-1$ blocks of M' bytes.” *Id.* at 2:41–43.

185. With respect to “wherein the shared memory allocated to the interleaver is used at the same time as the shared memory allocated to the deinterleaver,” Mazzoni discloses this limitation. As explained *supra*, I believe the appropriate construction of this term is as the district court construed it, i.e., “the deinterleaver reads from, writes to, or holds information for deinterleaving in its respective allocation of the shared memory at the same time as the interleaver reads from, writes to, or holds information for interleaving in its respective allocation of the shared memory.”

186. Mazzoni explains that “the interleaving means and the deinterleaving means include common memory means MM, e.g., a dual-port random access memory.” Ex. 1005 at 5:61–65. A POSA would have understood that a dual-port random access memory has two ports that would allow the disclosed interleaver to

read from or write to one port while the disclosed deinterleaver simultaneously reads from or writes to a second port. That is, by its disclosure of dual-port random access memory, Mazzoni discloses that the deinterleaver reads from, writes to, or holds information for deinterleaving in its respective allocation of the shared memory at the same time as the interleaver reads from, writes to, or holds information for interleaving in its respective allocation of the shared memory.

187. Mazzoni further explains that memory MM is “divided into a first memory space ESM1 assigned to the interleaving means MET, and a second memory space ESM2 [] assigned to the deinterleaving means MDET.” *Id.* at 5:64–67. By dividing the memory as disclosed, a POSA would have understood that the shared memory allocated to the interleaver is used at the same time as the shared memory allocated to the deinterleaver. Finally, a POSA would further have understood that where there is a constant stream of data being transmitted and received, for example received at 832x64 kbit/s in the remote unit and transmitted at 32x64 kbit/s from the remote unit, the deinterleaver would typically hold information for deinterleaving in its respective allocation of the shared memory at the same time as the interleaver holds information for interleaving in its respective allocation of the shared memory.

188. Mazzoni thus additionally discloses wherein the shared memory allocated to the interleaver (e.g., ESM1) is used at the same time (e.g., via the dual-

port random access memory) as the shared memory allocated to the deinterleaver (e.g., ESM2).

vii. Claim 2: The system of claim 1, wherein the determining is based on an impulse noise protection requirement.

189. It is my opinion that Mazzoni in view of VDSL1 discloses the system of claim 1, wherein the determining is based on an impulse noise protection requirement. For the system of claim 1, see *supra*, ¶¶92–187.

190. In disclosing how to compute the amount of interleaving or deinterleaving memory that must be allocated to an interleaver or deinterleaver, Mazzoni contemplates selecting interleaver and deinterleaver parameters based on a 0.25 millisecond burst of impulsive noise. *Id.* at 6:11–18. Indeed, this example disclosed by Mazzoni is premised on an impulse noise requirement, and thus the determining is based on said requirement. See ¶¶144-152.

191. VDSL1 also particularly discloses determining based on an impulse noise protection requirement. Specifically, VDSL1 explains “[i]nterleaving improves the error-correcting power of the RS codes in the presence of impulse

Table 27: Interleaver Characteristics

Parameter	Value	Notes
Block Length (I)	$I = N/8, N/4 \text{ or } N/2, \text{ octets}$	$N = PS+19, \text{ octets}$
Depth (D)	$D = M \times I + 1, \text{ octets}$	$M = 0 \rightarrow 64, \text{ programmable}$
Erasure Correction (E)	$E = \lfloor t \times I / N \rfloor \times (M \times I + 1), \text{ octets}$	$t = 8 \text{ (RS error correction ability)}$
End-to-End Delay (DL)	$DL = M \times I \times (I - 1), \text{ octets}$	
Interleaver Memory Size	$MEM = M \times I \times (I - 1) / 2, \text{ octets}$	
NOTE 1: The interleaver erasure correction E defines the maximum number of corrupted sequential octets could be corrected by RS algorithm when interleaving is applied. Correspondingly, the duration of noise pulses the system is protected from could be calculated as $E \times 8/R$, where R is the bit rate of the transmit signal over the line.		
NOTE 2: The $\lfloor \rfloor$ symbols indicates the truncation of the value to the lower integer.		

noise,” and that the “value of interleaving depth D shall be chosen in [accordance] with the required impulse noise protection (erasure correction).” Ex. 1007 at 71.

VDSL1 further explains how to calculate “Erasure Correction” in Table 27 (below) and adds that “the duration of noise pulses the system is protected from could be calculated as $Ex8/R$, where R is the bit rate of the transmit signal over the line.”

Id. at 72.

192. It likewise adds that the “maximum value of [parameter] M required for the given transmission profile shall provide an erasure correction capability up to 500 μs .” *Id.* at 72.

193. Also, Table B.1 shows how the determining is based on an impulse noise protection requirement.

Table B.1: Interleaving depth Parameter values

Line Rate, Mb/s		1,62	3,24	6,48	12,96	25,92	51,84
Value of N/I		8					
250 μs of erasure correction	M , octets	2	4	8	16	32	64
	Delay, msec	5,9					
500 μs of erasure correction	M , octets	4	8	16	32	64	128
	Delay, msec	11,8					

194. Consider Table B.1 for the example of $N=240$, which implies that $I = 30$ since Table B.1 indicates that $N/I = 8$. The table shown below uses the equation $M \times I \times (I-1)/2$ specified by VDSL1 to determine the memory required by the triangular implementation of convolutional interleaving for each set of parameters M and I . Note that the table shows how different amounts of memory are required

by VDSL1 for different impulse noise, for different latencies, and for different bit rates.

Impulse noise burst time in ms	0.25 ms			0.5 ms		
Corresponding latency in ms	5.9 ms			11.8 ms		
Bit Rate in Mb/s	M	I (for N=240)	Memory (bytes)	M	I for (N=240)	Memory (bytes)
1.62 Mb/s	2	30	870	4	30	1740
3.24 Mb/s	4	30	1740	8	30	3480
6.48 Mb/s	8	30	3480	16	30	6960
12.96 Mb/s	16	30	6960	32	30	13920
25.92 Mb/s	32	30	13920	64	30	27840
51.84 Mb/s	64	30	27840	128	30	55680

viii. Claim 3: The system of claim 1, wherein the determining is based on a latency requirement.

195. It is my opinion that Mazzoni in view of VDSL1 discloses the system of claim 1, wherein the determining is based on a latency requirement . For the system of claim 1, see above, ¶¶92–187.

196. Mazzoni explains that “[l]atency is a primordial and decisive criterion for any VDSL communication system.” Ex. 1005 at 2:54–56. Indeed, it touts the benefits of convolutional triangular interleaving in that it “reduces latency generated by the memory” because it “requires a much smaller memory, which reduces latency.” *Id.* at 2:49–56.

197. VDSL likewise discloses basing the determining of interleaver and deinterleaver memory on a latency requirement. For example, in its explanation of

interleaving and deinterleaving, it explains that the “interleaver and de-interleaver shall be adjustable via the management system to meet latency requirements.” Ex.

1007 at 59. It continues by explaining:

The latency of the slow path is a function of the data rate and burst error correction capability. For data rates greater than or equal to 13 Mbps, the latency between the α and β interfaces shall not exceed 10 ms when the interleaver delay is set to the maximum. At lower data rates there is a trade-off between higher latency and decreased burst error correction ability. At any data rate, the minimum latency occurs when the interleaver is turned off.

198. A POSA reading Mazzoni would have been motivated to look to VDSL1 as both Mazzoni and VDSL1 disclose the importance of latency requirements in interleaving and deinterleaving.

VDSL1 further contemplates various interleaver parameters that affect the determination of the interleaver and deinterleaver memory allocation, including “End-to-end delay,” i.e., latency (below,). Ex. 1007 at 72.

Table 27: Interleaver Characteristics

Parameter	Value	Notes
Block Length (I)	$I = N/8, N/4 \text{ or } N/2, \text{ octets}$	$N = PS+19, \text{ octets}$
Depth (D)	$D = M \times I + 1, \text{ octets}$	$M = 0 \rightarrow 64, \text{ programmable}$
Erasure Correction (E)	$E = \lfloor t \times I / N \rfloor \times (M \times I + 1), \text{ octets}$	$t = 8 \text{ (RS error correction ability)}$
End-to-End Delay (DL)	$DL = M \times I \times (I - 1), \text{ octets}$	
Interleaver Memory Size	$MEM = M \times I \times (I - 1) / 2, \text{ octets}$	
NOTE 1: The interleaver erasure correction E defines the maximum number of corrupted sequential octets could be corrected by RS algorithm when interleaving is applied. Correspondingly, the duration of noise pulses the system is protected from could be calculated as $E \times 8/R$, where R is the bit rate of the transmit signal over the line.		
NOTE 2: The $\lfloor \rfloor$ symbols indicates the truncation of the value to the lower integer.		

199. Also, Table B.1 shows how the determining is based on a latency requirement.

Table B.1: Interleaving depth Parameter values

Line Rate, Mb/s		1,62	3,24	6,48	12,96	25,92	51,84
Value of N/I		8					
250 μ s of erasure correction	M, octets	2	4	8	16	32	64
	Delay, msec	5,9					
500 μ s of erasure correction	M, octets	4	8	16	32	64	128
	Delay, msec	11,8					

200. Consider Table B.1 for the example of $N=240$, which implies that $I = 30$ since Table B.1 indicates that $N/I = 8$. The table shown below uses the equation $M \times I \times (I-1)/2$ specified by VDSL1 to determine the memory required by the triangular implementation of convolutional interleaving for each set of parameters M and I . Note that the table shows how different amounts of memory are required by VDSL1 for different impulse noise, for different latencies, and for different bit rates.

Impulse noise burst time in ms	0.25 ms			0.5 ms		
Corresponding latency in ms	5.9 ms			11.8 ms		
Bit Rate in Mb/s	M	I (for N=240)	Memory (bytes)	M	I for (N=240)	Memory (bytes)
1.62 Mb/s	2	30	870	4	30	1740
3.24 Mb/s	4	30	1740	8	30	3480
6.48 Mb/s	8	30	3480	16	30	6960
12.96 Mb/s	16	30	6960	32	30	13920
25.92 Mb/s	32	30	13920	64	30	27840
51.84 Mb/s	64	30	27840	128	30	55680

ix. Claim 4: The system of claim 1, wherein the determining is based on a bit error requirement.

201. It is my opinion that Mazzoni in view of VDSL1 discloses the system of claim 1, wherein the determining is based on a bit error requirement . For the system of claim 1, see above, ¶¶ 92–187.

202. Mazzoni discloses a Reed-Solomon coding means associated with an interleaver, which “can correct bursts of errors introduced by the transmission channel.” Ex. 1005 at 4:36–43. Mazzoni explains that, as an example, the “Reed-Solomon code is an RS (240,224) code with a correcting power of 8.” *Id.* at 4:44–48. “This notation means that the Reed-Solomon coding means are applied to packets of 224 bytes, to which they add 16 parity bytes, to form a Reed-Solomon coded word whose length is 240 bytes,” which “makes it possible to correct up to eight erroneous bytes.” *Id.* at 4:48–52. Therefore, the determination of the amount of memory required by the interleaver or deinterleaver is based on a bit error requirement.

203. Mazzoni further explains that the errors introduced into the channel may exceed the correction capacity of the Reed-Solomon code, and thus may have to be redistributed by temporally interleaving them. Such a step makes clear that the interleaver memory would need to be set according to a bit error requirement. Specifically, Mazzoni discloses:

Errors introduced by the channel, which often occur in bursts affecting several successive bytes and can therefore exceed the correction capacity of the Reed-Solomon code in isolation, may be distributed temporally. To do so, the bytes are temporally interleaved by modifying the order in which

they are transmitted. This improves the efficacy of the Reed-Solomon coding.

Id. at 4:53–59.

204. VDSL1 likewise discloses the determination of the amount of memory allocated to the interleaver to be based on a bit error requirement. Indeed, VDSL explains that bit error rate and latency are intertwined. Ex. 1007 at 59. Thus, one is left balancing the latency requirements with bit error requirements. Specifically, VDSL1 explains:

The latency of the slow path is a function of the data rate and burst error correction capability. For data rates greater than or equal to 13 Mbps, the latency between the α and β interfaces shall not exceed 10 ms when the interleaver delay is set to the maximum. At lower data rates there is a trade-off between higher latency and decreased burst error correction ability. At any data rate, the minimum latency occurs when the interleaver is turned off.

Id.

205. A POSA reading Mazzoni would have been motivated to look to VDSL1 as both Mazzoni and VDSL1 disclose the importance of bit error requirements in interleaving and deinterleaving.

XIII. Ground 2: The combination of VDSL1 and Fadavi-Ardekani renders the challenged claims obvious.

b. Analysis

i. Claim 1: [1.pre] A system that allocates shared memory comprising:

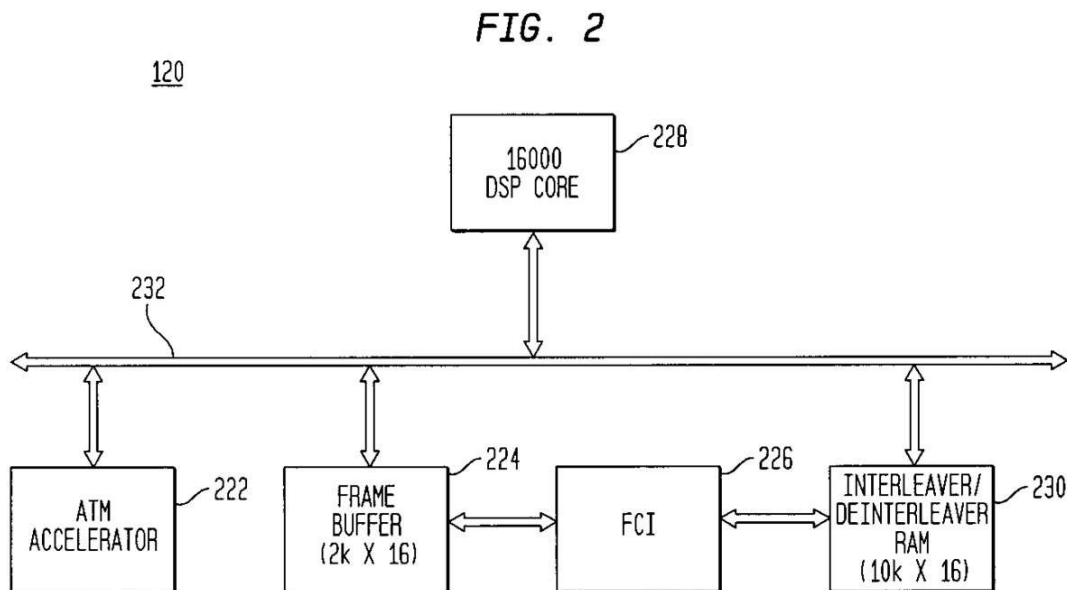
206. This claim element contains the claim term “shared memory,” which I understand was previously construed in a prior litigation as “common memory used by at least two functions, where a portion of the memory can be used by either one of the functions.” In my opinion and in view of this construction, Fadavi-Ardekani explicitly discloses “a system that allocates shared memory.”

207. I understand that, in the prior Delaware litigation, the district court granted summary judgment of no invalidity determining that Fadavi-Ardekani did not directly disclose a shared memory as construed—namely, a shared memory *where a portion of the memory can be used by either one of the functions*—and that the Defendants failed to assert that such a shared memory was inherent in the disclosure. *2Wire*, No. 13-cv-01835-RGA, ECF. No. 1106, at 17. I disagree with such a determination as, in my view, Fadavi-Ardekani expressly and inherently discloses a shared memory as construed by the district court.

208. For example, Fadavi-Ardekani discloses an “Interleave/De-Interleave Memory (IDIM)” that “provides a memory” and “may be utilized in a ping-pang fashion.” Ex. 1006 at 6:55–58. Fadavi-Ardekani defines “ping-pang” to mean that “areas of the memory buffer are alternately utilized exclusively by one agent (a transceiver component for performing some function) and then by a second agent” and that “[a]s one area of memory is being used by a first agent, another area of memory can be used by a different agent” and “[a]t any time, an agent is allowed

to access either a ping area of memory or a pang area of memory.” *Id.* at 5:60–6:3.

Fadavi-Ardekani provides Figure 2, which identifies an “Interleaver/Deinterleaver RAM” (below).



209. It is my opinion that the IDIM disclosed by Fadavi-Ardekani is a “shared memory” as construed by the District of Delaware. The IDIM is a common memory used by at least two functions, namely interleaving and deinterleaving. In fact, Fadavi-Ardekani teaches using the same common IDIM memory to support multiple interleavers and deinterleavers so that multiple transceivers can utilize this memory. Fadavi-Ardekani teaches that the transceivers, and necessarily the common IDIM memory, support various xDSL communications systems including VDSL, as shown in the excerpt below:

the buffering and Scheduling Scheme of the invention may also be utilized in transceivers for various xDSL communication systems including High bit-rate DSL (HDSL) and Very high bit rate DSL (VDSL). *Id.* at 8:18-21.

210. Fadavi-Ardekani teaches that a portion of the memory can be used for either interleaving or deinterleaving when he discloses IDIM can be used for either G.Lite sessions or standard ADSL sessions, each with distinct interleaver/deinterleaver memory requirements. These disclosures are described, for example, in the excerpt below:

In a preferred embodiment of the invention, the IDIM is allocated as 10 Kx16 (i.e., 20 K) Random Access Memory (RAM), which supports approximately four G.lite or approximately four Standard ADSL Session/s at less than full interleave depth. The size of the IDIM and the interleave depth may be varied so that a different number of sessions may be supported by the transceiver of the invention. *Id.* at 6:66-7:5

211. Specifically, Fadavi-Ardekani teaches two implementations of allocated shared memory: (1) a “simple implementation”; and (2) an “optimal implementation.”

212. In the simple implementation, the required IDIM size for handling G.lite ADSL session is computed as the sum of the memory for downstream processing:

The size of the IDIM is derived as follows. A simple implementation of a transmit interleaver for G.lite communication requires 4 Kbytes per session for downstream processing, derived by multiplying the maximum codeword length by the maximum interleaver

depth. The simple G.lite transmit interleaver also requires 2 Kbytes per Session for upstream processing. Therefore, 24 Kbytes of RAM is required to support four G.lite Sessions. Similarly, a simple implementation of a transmit interleaver for standard ADSL requires 16 Kbytes per session for downstream processing and 2Kbytes per Session for upstream processing, for a total of 72 Kbytes for four sessions. A fast path buffer is also required for fast path data in both the interleave and de-interleave processes and requires 256 bytes of RAM per Session, or a total of 1 K bytes for four sessions. Since the smallest RAM block currently available is 1 Kx 16, 1 Kx 16 or 2 Kbytes must be allocated for the fast path buffer per direction. Therefore, a simple implementation of an interleaver would require 76 Kbytes for four standard ADSL sessions (64 Kinterleave +8K de-interleave +4 K fast path). *See id.* at 7:5-25.

213. Considering only the simple implementation of an ADSL session, Fadavi-Ardekani discloses that it requires “16 Kbytes per session for downstream processing” and “2 Kbytes per session for upstream processing,” thus requiring 18 Kbytes of RAM per session, or “72 Kbytes for four sessions.” *Id.* at 7:13–17. Thus, in the simple implementation, where the RAM size is equivalent to the maximum possible sum of interleaving and deinterleaving requirements for all sessions, i.e., 18 Kbytes for one session or 72 Kbytes for four sessions, and the *ratio of interleaver and deinterleaver memory sizes are fixed*, it may be implementable in either a shared memory (i.e., where there may be byte overlap between functions) or in a dedicated memory (i.e., where there will not be byte overlap between functions) because there is enough capacity to implement each 18

Kbyte session without using some bytes for both interleaving and deinterleaving. Even though separate dedicated memories could be contemplated in this specific fixed-ratio situation, Fadavi-Ardekani teaches away from such needless rigidity by using a single random access memory, which inherently allows any function to access any byte of memory by using the appropriate address. Each time the memory is reallocated, the user could allocate the interleavers and deinterleavers to different regions of memory than previous allocations with no perceivable difference in performance. When Fadavi teaches that the memory sizes of the deinterleavers and interleavers grow and shrink with their parameters such as interleaver depth, *id.* 7:30-32, a POSA would have understood that sections of RAM are being reallocated without any attention to needlessly ensuring that bytes of RAM used for interleaving in previous allocations is still used for interleaving rather than deinterleaving in a new allocation.

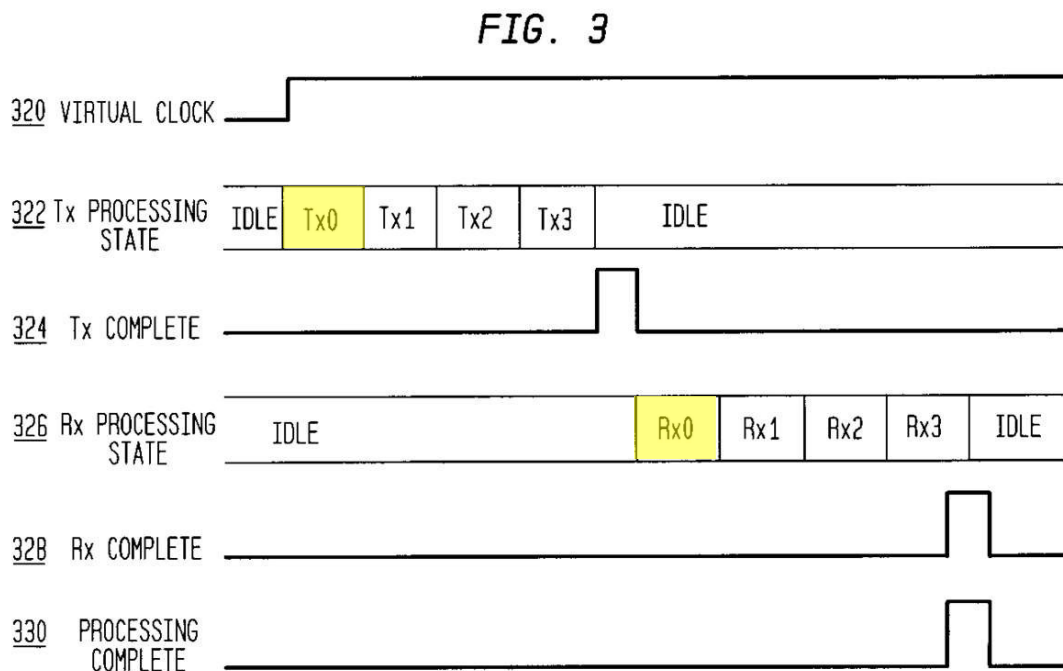
214. Furthermore, a POSA looking to Fadavi-Ardekani for a beneficial environment in which to implement VDSL would have understood that a shared memory would be selected because support of both asymmetric and symmetric streams requires that *the ratio of interleaver and deinterleaver sizes is not fixed* and the flexibility taught by Fadavi-Ardekani to use portions of the memory for interleaving or deinterleaving of various sessions as needed would necessarily be required to implement the full range of VDSL. Ex. 1006 at 7:30–33. Specifically,

VDSL1 contemplates both symmetric and asymmetric systems, Ex. 1007 at 37, meaning that, when using the full capacity of the RAM, there would be some bytes of memory used for interleaving in the office transceiver in an asymmetric implementation that are also used for deinterleaving in the office transceiver in a symmetric implementation.

215. Regardless, in the “optimal implementation of the interleaver,” Fadavi-Ardekani expressly discloses a shared memory where a portion of the memory can be used by either one of the functions. Fadavi-Ardekani explains that the invention “utilizes the same memory for receive data and transmit data,” and thus the interleaver/deinterleaver may be assigned only 16 Kbytes. Ex. 1006 at 7:25–30. Because one ADSL session requires 16 Kbytes in the downstream and 2 Kbytes in the upstream—for a total of 18 Kbytes—and the RAM available to the interleaver/deinterleaver is only 16 Kbytes (after accounting for 4 Kbytes for the fast path buffer), a portion of the memory must be used by both the interleaver and deinterleaver to implement the disclosed session.

216. Specifically, looking at Figure 3, below, Fadavi-Ardekani discloses that the FCI “first processes TX data for each active ADSL line,” which may include the 16 Kbytes of interleaved data (highlighted below, Tx0), and then “processes RX data for each active ADSL line,” which may include 2 Kbytes of

deinterleaved data (highlighted below, Rx0), all in a single session that was originally only allocated 16 Kbytes of RAM.



217. Claim 12 of Fadavi-Ardekani also discloses “common memory used by at least two functions, where a portion of the memory can be used by either one of the functions” when it teaches an IDIM that “is a RAM of a size that supports at least four full depth G.lite sessions or approximately one full depth standard ADSL session.” Ex. 1006 at 10:52–55. Here, for the simple implementation, the standard ADSL session requires 16 Kbytes for the interleaving of the downstream RS coded bytes by the central office transceiver, and 2K bytes deinterleaving of the upstream RS coded bytes by the central office transceiver. Each G.lite session requires 4

Kbytes for the interleaving of the downstream RS coded bytes by the central office transceiver and 2K bytes deinterleaving of the upstream RS coded bytes by the central office transceiver. Thus, a POSA would have understood that any byte of the RAM that is in the IDIM can be used for interleaver or for deinterleaving to accommodate each of these sessions for either the G.lite or for the standard ADSL.

218. At a minimum, in the optimum implementation, Fadavi-Ardekani discloses a common memory (e.g., the IDIM) used by at least two functions (e.g., the interleaver and the deinterleaver), where a portion of the memory (e.g., 16 Kbytes of RAM) can be used by either one of the functions.

**ii. Claim 1: [1.A] a transceiver that is capable of:
transmitting or receiving a message during initialization
specifying a maximum number of bytes of memory that are
available to be allocated to an interleaver;**

219. This claim element contains the claim term “transceiver,” which I understand was previously construed in a prior litigation as “communications device capable of transmitting and receiving data wherein the transmitter portion and receiver portion share at least some common circuitry.” In my opinion, the plain and ordinary meaning of this term as known to a POSA would be a “communications device capable of transmitting and receiving data.” Nonetheless, in my opinion and in view of the prior construction, the combination of VDSL1 and Fadavi-Ardekani explicitly discloses a “a transceiver that performs: transmitting or receiving a

message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver.”

220. VDSL1 expressly discloses “a transceiver that performs: transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver.”

221. VDSL1 discloses “a transceiver that performs: transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver.” Specifically, the VDSL1 standard both discloses a transceiver and a message that is transmitted from the VTU-R transceiver and received by the VTU-O transceiver that explicitly specifies a maximum number of bytes of memory that are available to be allocated to an interleaver. *See* Ex. 1007 at 127, 132.

222. First, the VDSL1 standard discloses a transceiver. The standard explicitly refers to its communication devices as transceivers. For example, it defines the acronym “VTU,” which is used throughout the standard, to mean “VDSL Transceiver Unit.” Ex. 1007 at 13. Also defined are the acronyms “VTU-O,” which means the VTU located at the Optical Network Unit (ONU) or operator side, and VTU-R, which means the VTU located at the Remote site, which is the Network Termination (NT) or user side. *Id.*

223. Next, each VTU is capable of transmitting and receiving data. Figure 4 from the VDSL1 technical specification, shown below, illustrates the reference configuration used for the VDSL1 technical specification. The figure shows how VTU-R transceivers, located at the NT, transmit and receive data to and from the VTU-O, which is located at the ONU. Ex. 1007 at 16. Similarly, the VTU-O transceivers transmit and receive data to and from the VTU-R. *Id.*

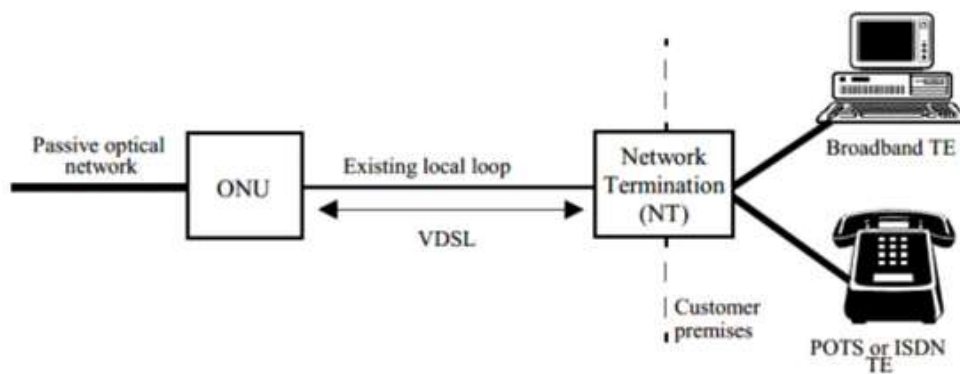


Figure 4: General fibre-to-the-node architecture for VDSL

224. The transmitter and receiver portions of the VTUs share some circuitry. For example, such devices share the circuitry that provides power. Ex. 1007 at 20. Also, the configuration messages that are received by the VTU are used to affect both transmitter and receiver functionality, so the circuitry that utilizes the messages is shared by the transmitter and receiver portions of the VTU. *Id.* at 111. The transmitting and receiving functions commonly share the same clock to control the timing of the digital circuitry. *Id.*

225. In addition to disclosing a transceiver, the VDSL1 standard discloses transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver. Specifically, the VDSL1 standard discloses how each of its transceivers transmit or receive a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver in the “channel analysis and exchange state” of the “initialization protocol.” Ex. 1007 at 111.

226. Figure 50, shown below, shows the “Channel analysis and exchange” state as part of initialization. *Id.*

VTU-O		
Activation: Handshake procedures (clause 8.2.3)	Training (clause 8.2.4)	Channel analysis and Exchange (clause 8.2.6)
VTU-R		
Activation: Handshake procedures (clause 8.2.3)	Training (clause 8.2.4)	Channel analysis and Exchange (clause 8.2.6)

Figure 50: Overview of initialisation

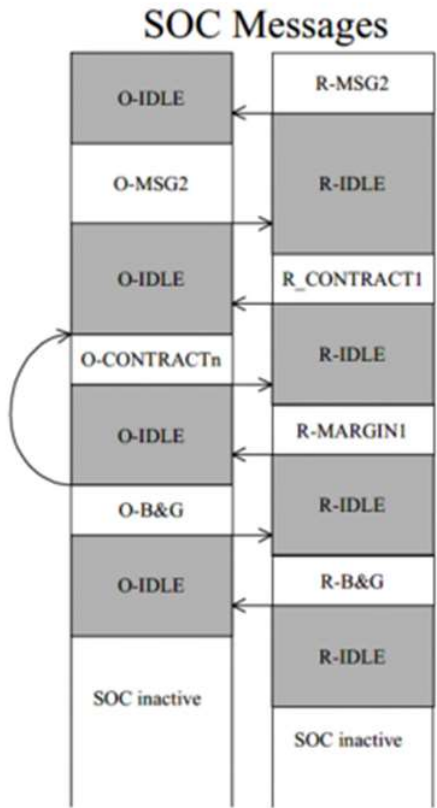
227. A POSA would have understood that the description of Figure 50 discloses that during the channel analysis and exchange state, the VTU-O and VTU-R measure the characteristics of the channel and agree on all the parameters needed to thoroughly define the communication link. Specifically, VDSL1 discloses:

The time line in Figure 50 provides an overview of the initialisation protocol. Following the initial handshake procedure, a full duplex link between the VTU-O and the VTU-R is established. During the training phase, timing advance and upstream power back-off shall be refined. During the channel analysis and exchange state, the two modems

measure the characteristics of the channel and agree on a contract that thoroughly defines the communication link.

Id. at 111.

228. Figure 56 provides an overview of the messages exchanged during the channel analysis and exchange state, which are used to agree on all the parameters needed to thoroughly define the communication link.



Id. at 127 (Excerpt from Figure 56 showing messages exchanged during the “Channel analysis and exchange” state).

229. During the “Channel analysis and exchange” state, the VTU-R sends a specific message, R-MSG2, to the VTU-O to describe its capabilities, including its

maximum available interleaver memory. *Id.* at 132. For example, the message R-MSG2 includes the field “Maximal interleaver memory,” which specifies the maximal number of bytes available to be allocated to an interleaver. *Id.* at 132. The R-MSG2 contains information about the capabilities of the VTU-R for bit allocation. *Id.* The content of R-MSG2 is specified in Table 82, which is shown below.

Table 82: Description of R-MSG2

Field	Field or macro-field type	Remark
Message descriptor	1 octet	See table 49
Maximum constellation size in upstream	1 octet	Maximum number of bits per tone
RS setting supported by VTU-R	1 octet	0x00: only mandatory settings 0xFF: all settings (see note 1)
Interleaver setting supported by VTU-R	1 octet	0x00: only mandatory settings 0xFF: all settings 0xNN: NN = number of settings (0x00 < NN < 0xFF)
Detailed interleaver setting description	0 octets if NN = 0x00 0 octets if NN = 0xFF NN x 4 octets otherwise	See table 74
Maximum power transmitted	1 octet	In units of 0,25 dBm
Maximum interleaver memory	3 octets	In octets (see note 2)
Maximum number of EOC octets per frame in upstream	1 octet	Number of EOC octets per frame
Maximum number of VOC octets per frame in upstream	1 octet	Number of VOC octets per frame
Support of express swapping	1 octet	0x00: Not supported 0xFF: Supported
j_{max}	1 octet	Maximum value of j_{max} supported by the VTU-R (see note 3)
NOTE 1: All settings means the values (for the redundancy) that are specified in clause 6.4.2.		
NOTE 2: The interleaver memory is computed as $M \times l \times (l-1)$.		
NOTE 3: Specification of $j_{max} = k$ means that all values from 0 to k are supported.		

Id.

230. After receiving this message, the VTU-O sends the message O-MSG2 describing its capabilities. *Id.* at 127. After receiving the O-MSG2, the VTU-R sends message R-CONTRACT1 to the VTU-O to propose parameters for the downstream channel Reed-Solomon coding and interleaving, within the

capabilities as set forth in the R-MSG2 and O-MSG2 messages. *Id.* at 132. After receiving R-CONTRACT1, the VTU-O responds with its proposed parameters for both upstream and downstream channels using the O-CONTRACT1 message. *Id.* at 129. The O-CONTRACT1 message bases the downstream contract on the interleaving parameters proposed in R-CONTRACT1. Ideally the downstream contract is the same as the one proposed in R-CONTRACT1. *Id.* at 129. A POSA would have understood that the upstream contract proposed by the VTU-O must comport with the limitations of the VTU-R including maximum interleaver memory as communicated in R-MSG2.

231. When the VTU-R receives the O-CONTRACT1 message, it computes the minimal SNR margin over all tones and sends this information to the VTU-O in the R-MARGIN1 message. *Id.* at 132. If the VTU-O finds the SNR margins in the R-MARGIN1 message acceptable, then it sends the O-B&G message, which ends the negotiation. *Id.* at 129–30. If the VTU-O does not find the SNR margins in the R-MARGIN1 message acceptable, then it sends a new O-CONTRACT2 message and the VTU-R computes the new values for the minimal SNR margin over all tones and sends this information to the VTU-O in the R-MARGIN2 message. *Id.* at 129–32. The VTU-O and VTU-R continue to exchange O-CONTRACTn and the R-MARGINn messages until the VTU-O is satisfied and sends the O-B&G message which ends the negotiation. *Id.*

232. A POSA would have understood that, throughout the exchange, the VTU-O would not propose a contract that exceeds the available interleaver memory at the VTU-R, which is described in the “maximum interleaver memory” field of the R-MSG2 sent by the VTU-R.

233. Fadavi-Ardekani likewise discloses a transceiver, i.e., a communication device capable of transmitting and receiving data (e.g., a “transceiver for an asymmetric communication system”). Ex. 1006 at Abstract. Within the Detailed Description, under the heading “Central Office ADSL Transceiver” Fadavi-Ardekani states that “Fig. 2 illustrates a central office ADSL transceiver according to the invention.” *Id.* at 4:64-67. As described with respect to Limitation [1.pre], Fadavi-Ardekani further discloses a shared memory “IDIM,” which is “bi-directionally coupled to [a framer/coder/interleaver (FCI)].” *Id.* at 9:49–55. The POSA would have understood the shared memory of Fadavi-Ardekani is common circuitry between the “transmitters and receivers.” *Id.* at Abstract (“Memory is shared by multiple ADSL sessions and by the transmit and receive processes within an individual session.”).

234. A POSA would have understood that Fadavi-Ardekani and the VDSL1 standard both disclose techniques and capabilities for the same goal of VDSL communication, and that the messages exchanged during the “Channel analysis and exchange” state are easily combined with the transceivers described

by Fadavi-Ardekani. Indeed, the set of messages defined by VDSL1 for the “Channel analysis and exchange” state provide a benefit to the VDSL transceivers disclosed by Fadavi-Ardekani by allowing VDSL transceivers at the operator end (TO or VTU-O) and the user end (TU or VTU-R) to exchange information about their capabilities and agree on all of the parameters needed to thoroughly define the communication link.

235. Therefore, it is my opinion that a POSA would have understood that the combination of VDSL1 and Fadavi-Ardekani disclose “a transceiver that performs: transmitting or receiving a message during initialization specifying a maximum number of bytes of memory that are available to be allocated to an interleaver.

iii. Claim 1: [1.B] determining an amount of memory required by the interleaver to interleave a first plurality of Reed Solomon (RS) coded data bytes within the shared memory;

236. This claim element contains the term “amount of memory,” which I understand was previously construed as the plain meaning without being limiting to a number of bytes. In my view, a POSA would have understood the plain and ordinary meaning of the term “an amount of memory,” without further explanation.

237. In my opinion, the combination of VDSL1 and Fadavi-Ardekani discloses determining an amount of memory required to interleave a first plurality of Reed Solomon (RS) coded data bytes within a shared memory.

238. The VDSL1 technical specification describes determining an amount of memory required by the interleaver to interleave a first plurality of Reed Solomon (RS) coded data bytes within a shared memory. For example, VDSL1 explains that the “(De)interleaver memory size” can be determined from the equation $M \times I \times (I-1)/2$ (see Table 19 below), wherein parameter I is the interleaver block length and parameter M is the interleaving depth parameter. *Id.* at 60.

Table 19: Characteristics of triangular, convolutional interleaver

Parameter	Value
Interleaver block length	I octets (I must divide N)
Interleaving depth D	$M \times I + 1$ octets
(De)interleaver memory size	$M \times I \times (I-1)/2$ octets
End-to-end delay	$M \times I \times (I-1)$ octets
Correction capability	$\lfloor t/q \rfloor \times M \times (I+1)$ octets

239. VDSL1 also contemplates interleaving Reed-Solomon codewords, explaining that “[i]nterleaving shall be used to protect the data against bursts of errors by spreading the errors over a number of Reed-Solomon codewords,” and “[w]hen the interleaver is on, the codewords shall be interleaved before transmission to increase the immunity of RS codewords to bursts of errors.” *Id.* at 59. VDSL1 further requires that interleavers be supported for Reed-Solomon codes. *Id.* at 59–62. Specifically, VDSL1 explains that a “standard octet-oriented Reed-Solomon code shall be used to provide protection against random and burst errors,” where a “Reed-Solomon code word contains $N=K+R$ octets,” and at least (N,K) values of (144,128) and (240,224) must be supported. *Id.* at 59.

240. The VDSL1 technical specification indicates that various choices of the interleaver M and I may be employed, for example, as a function of the bit rate (Line Rate) and the impulse noise protection and the corresponding latency (Delay) Mb/s. Using I —the interleaver block length—and M —the interleaving depth index—the interleaver or deinterleaver memory is determined. *Id.* at 71–72. For example, the technical specification provides an example explaining that some “[s]ome typical examples of interleaving parameters values of M , E and end-to-end delay calculated for $N/I = 8$, $t = 8$ and different line rates are presented in table B.1.” *Id.* at 153.

Table B.1: Interleaving depth Parameter values							
Line Rate, Mb/s		1,62	3,24	6,48	12,96	25,92	51,84
Value of N/I		8					
250 μs of erasure correction	M, octets	2	4	8	16	32	64
	Delay, msec	5,9					
500 μs of erasure correction	M, octets	4	8	16	32	64	128
	Delay, msec	11,8					

241. Consider Table B.1 for the example of $N=240$, which implies that $I = 30$ since Table B.1 indicates that $N/I = 8$. The table shown below uses the equation $M \times I \times (I-1)/2$ specified by VDSL1 to determine the memory required by the triangular implementation of convolutional interleaving for each set of parameters M and I . Note that the table shows how different amounts of memory are required by VDSL1 for different impulse noise, for different latencies, and for different bit rates.

Impulse noise burst time in ms	0.25 ms			0.5 ms		
Corresponding latency in ms	5.9 ms			11.8 ms		
Bit Rate in Mb/s	M	I (for N=240)	Memory (bytes)	M	I for (N=240)	Memory (bytes)
1.62 Mb/s	2	30	870	4	30	1740
3.24 Mb/s	4	30	1740	8	30	3480
6.48 Mb/s	8	30	3480	16	30	6960
12.96 Mb/s	16	30	6960	32	30	13920
25.92 Mb/s	32	30	13920	64	30	27840
51.84 Mb/s	64	30	27840	128	30	55680

242. Relying on the disclosed equation $M \times I \times (I-1)/2$, a POSA would have understood how to determine the “(De)interleaver memory size.” *Id.* at 60, 71–72. A POSA would further make this determination for a shared memory—such as the shared memory disclosed in Fadavi-Ardekani, IDIM—as disclosed in [1.pre] for the benefits described *supra*.

iv. Claim 1: [1.C] allocating a first number of bytes of the shared memory to the interleaver to interleave the first plurality of Reed Solomon (RS) coded data bytes for transmission at a first data rate, wherein the allocated memory for the interleaver does not exceed the maximum number of bytes specified in the message;

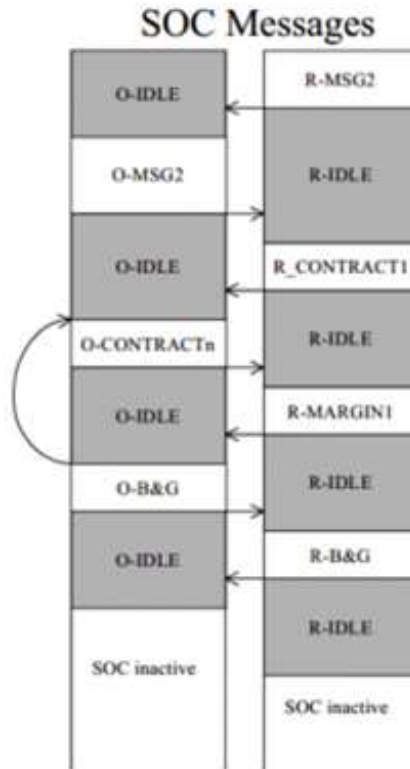
243. In my opinion, the combination of VDSL1 and Fadavi-Ardekani disclose this limitation for the same reasons as discussed in Limitations [1.A] and [1.B]. See *supra* ¶¶227–250. Specifically, VDSL1 and Fadavi-Ardekani disclose allocating a first number of bytes of the shared memory (in Fadavi-Ardekani, the IDIM) to the interleaver to interleave the first plurality of Reed Solomon (RS)

coded data bytes (in VDSL1, as described *supra*) for transmission at a first data rate (in VDSL1, as described *supra*, e.g., 51.84 Mbps).

244. In addition to *determining* an amount of memory required to interleave a first plurality of Reed-Solomon coded data bytes as described *supra*, VDSL1 discloses *allocating* the required amount to the interleaver.

245. VDSL1 discloses a transceiver and a message that is transmitted from the VTU-R transceiver and received by the VTU-O transceiver that explicitly specifies a maximum number of bytes of memory that are available to be allocated to an (de)interleaver at initialization. *See* Ex. 1007 at 132; *see also* [1.A] and [1.B], *supra*.

246. As explained *supra*, VDSL1 discloses that in the “initialization protocol,” there is a “channel analysis and exchange state” whereby the modems “agree on a contract that thoroughly defines the communication link.” Ex. 1007 at 111. As depicted in excerpted Figure 56, the “channel analysis and exchange state” includes a number of messages delivered between the VTU-O and VTU-R.



Ex. 1007 at 127.

247. As explained *supra*, the VTU-R and VTU-O begin a series of negotiations to set the parameters for downstream and upstream interleaving and deinterleaving. Specifically, VDSL1 explains that the “VTU-R sends the R-MSG2 message to transfer information about its bit allocation capabilities.” Ex. 1007 at 127. After receiving R-MSG2, the VTU-O sends the message O-MSG2 describing its capabilities. Ex. 1007 at 128. After receiving the O-MSG2, the VTU-R sends message R-CONTRACT1 to the VTU-O to propose parameters for the downstream channel Reed-Solomon coding and interleaving, within the

capabilities as set forth in the R-MSG2 and O-MSG2 messages. *Id.* at 131. After receiving R-CONTRACT1, the VTU-O responds with its proposed parameters for both upstream and downstream channels using the O-CONTRACTn message. *Id.* at 129.

248. As depicted in Table 75, the O-CONTRACTn, delivers a contract descriptor message for the downstream contract and upstream contract. *Id.* at 129. This message coming from the VTU-O includes bits specifying the values of M and I as the interleaver or deinterleaver settings. *Id.*

Table 75: Description of O-CONTRACTn

Field	Field or macro-field type
Message descriptor	Message code (1 octet)
Downstream contract	Contract Descriptor
Upstream contract	Contract Descriptor
EOC capacity	Number of EOC octets per frame (1 octet)
VOC capacity	Number of VOC octets per frame (1 octet)

249. As depicted in Table 76, the contract descriptor message for the downstream contract includes the parameters I and M for the interleaver (here, labeled “Interleaver setting”). Ex. 1007 at 129.

Table 76: Contract Descriptor

Field	Field or macro-field type	Remark
Rate in fast channel	2 octets	In multiple of 64 kbps
RS setting in fast channel	2 octets	b15 → b8: RS overhead b7 → b0: RS codeword length
Rate in slow channel	2 octets	In multiple of 64 kbps
RS setting in slow channel	2 octets	b15 → b8: RS overhead b7 → b0: RS codeword length
Interleaver setting	2 octets	b15 → b8: M (see note 1) b7 → b0: I
NOTE: The value I must be a divider of the RS codeword length.		

250. A POSA would have understood that the message delivered by the final O-CONTRACT_n message, which included the I and M parameters from which the memory will be determined according to the equation specified in VDSL1, is the allocation of that memory. Indeed, having set the parameters I and M during initialization, VDSL1 discloses, in Table 19, that the interleaving and deinterleaving memory size may be determined, and subsequently allocated, based on the parameters I and M using equation $M \times I \times (I-1)/2$. Ex. 1007 at 60.

Table 19: Characteristics of triangular, convolutional interleaver

Parameter	Value
Interleaver block length	I octets (I must divide N)
Interleaving depth <i>D</i>	$M \times I + 1$ octets
(De)interleaver memory size	$M \times I \times (I-1)/2$ octets
End-to-end delay	$M \times I \times (I-1)$ octets
Correction capability	$\lfloor t/q \rfloor \times M \times (I+1)$ octets

251. The O-CONTRACT₁ message bases the downstream contract on the interleaving parameters proposed in R-CONTRACT₁. Ideally the downstream contract is the same as the one proposed in R-CONTRACT₁. *Id.* at 129. A POSA would have understood that the upstream contract proposed by the VTU-O must comport with the limitations of the VTU-R including maximum interleaver memory as communicated in R-MSG2. As such, a POSA would have understood that the O-CONTRACT_n message confirms that the allocated memory for the interleaver does not exceed the maximum number of bytes specified in the R-MSG2 message. For example, the memory allocation for the upstream interleaver

in the VTU-R does not exceed the maximum number of bytes specified in the R-MSG2 sent by the VTU-R.

252. The combination of VDSL1 and Fadavi-Ardekani thus also discloses wherein the allocated memory for the interleaver (in VDSL1, set by message O-CONTRACTn) does not exceed the maximum number of bytes specified in the message (in VDSL1, set by message R-MSG2).

v. Claim 1: [1.D] allocating a second number of bytes of the shared memory to a deinterleaver to deinterleave a second plurality of RS coded data bytes received at a second data rate; and

253. VDSL1 and Fadavi-Ardekani disclose this limitation for the same reasons as discussed in Limitation [1.C]. See *supra*, ¶¶251–260. Indeed, VDSL1 discloses various “*(De)*interleaver memory size[s],” indicative of the fact that a second number of bytes of the shared memory may be allocated to a deinterleaver when the first number of bytes was allocated to an interleaver.

254. Likewise, although I understand the claim does not require that the second data rate be different from the first data rate, as I explained in Limitation [1.B], the determination of memory to be allocated may be based on various parameters, including “different line rates.” Ex. 1007 at 153. Therefore, a POSA would have understood that the allocation of memory to a deinterleaver to

deinterleave RS coded data bytes may be based on a data rate different than that of data rate used for allocating memory to an interleaver.

255. The combination of VDSL1 and Fadavi-Ardekani thus disclose allocating a second number of bytes of the shared memory to a deinterleaver to deinterleave a second plurality of RS coded data bytes received at a second data rate.

vi. Claim 1: [1.E] interleaving the first plurality of RS coded data bytes within the shared memory allocated to the interleaver and deinterleaving the second plurality of RS coded data bytes within the shared memory allocated to the deinterleaver, wherein the shared memory allocated to the interleaver is used at the same time as the shared memory allocated to the deinterleaver.

256. As is discussed in Limitations [1.B]–[1.D], *supra*, the combination of VDSL1 and Fadavi-Ardekani disclose interleaving the first plurality of RS coded data bytes within the shared memory allocated to the interleaver and deinterleaving the second plurality of RS coded data bytes within the shared memory allocated to the deinterleaver.

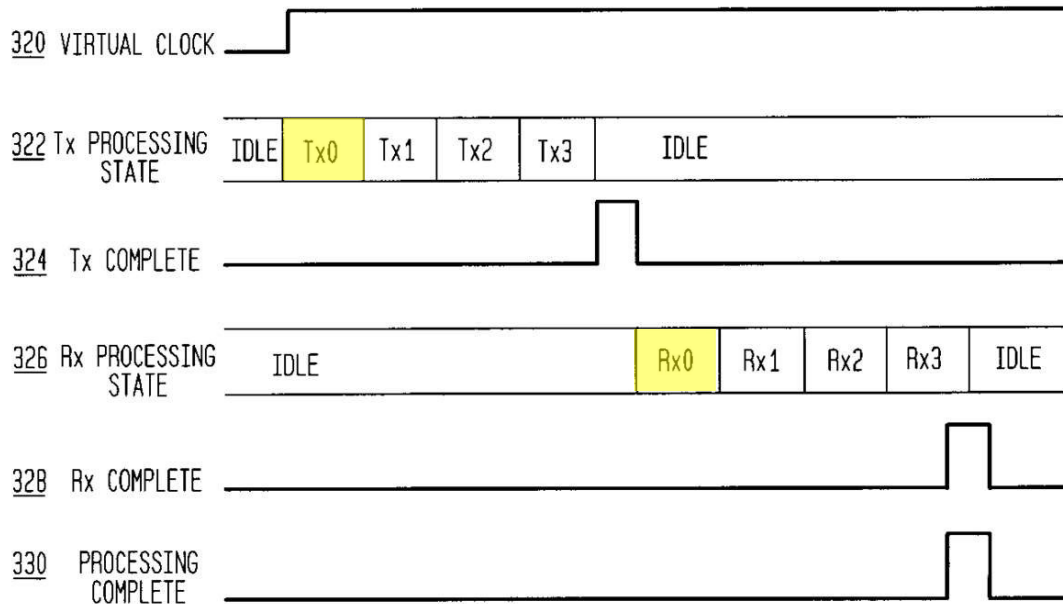
257. With respect to “wherein the shared memory allocated to the interleaver is used at the same time as the shared memory allocated to the deinterleaver,” Fadavi-Ardekani discloses this limitation. As explained *supra*, it is my opinion that the district court’s construction is correct, namely that “the shared memory allocated to the [deinterleaver / interleaver] is used at the same time as the

shared memory allocated to the [interleaver / deinterleaver]” means “the deinterleaver reads from, writes to, or holds information for deinterleaving in its respective allocation of the shared memory at the same time as the interleaver reads from, writes to, or holds information for interleaving in its respective allocation of the shared memory.”

258. Fadavi-Ardekani discloses an “Interleave/De-Interleave Memory (IDIM)” that “provides a memory” and “may be utilized in a ping-pang fashion.” Ex. 1006 at 6:55–58. Fadavi-Ardekani defines “ping-pang” to mean that “areas of the memory buffer are alternately utilized exclusively by one agent (a transceiver component for performing some function) and then by a second agent” and that “[a]s one area of memory is being used by a first agent, another area of memory can be used by a different agent.” *Id.* at 5:60–65. A POSA would have understood that use of the memory in a ping-pang fashion would allow the disclosed interleaver to read from or write to one port while the disclosed deinterleaver simultaneously reads from or writes to a second port.

259. Likewise, in Figure 3, below, Fadavi-Ardekani also discloses that the FCI “first processes TX data for each active ADSL line,” (i.e., writes to) and then “processes RX data for each active ADSL line,” (i.e., reads from) all in a single session. Ex. 1006 at 7:63–8:4.

FIG. 3



260. The combination of VDSL1 and Fadavi-Ardekani thus discloses this limitation, including wherein the shared memory allocated to the interleaver (e.g., in IDIM) is used at the same time (e.g., in a ping-pang fashion) as the shared memory allocated to the deinterleaver (e.g., in IDIM).

vii. Claim 2: The system of claim 1, wherein the determining is based on an impulse noise protection requirement.

261. It is my opinion that VDSL1 in view of Fadavi-Ardekani discloses the system of claim 1, wherein the determining is based on an impulse noise protection requirement. For the system of claim 1, see above, ¶¶ 214–268.

262. VDSL1 particularly discloses determining based on an impulse noise protection requirement. Specifically, VDSL1 explains “[i]nterleaving improves the error-correcting power of the RS codes in the presence of impulse noise,” and that the “value of interleaving depth D shall be chosen in [accordance] with the required impulse noise protection (erasure correction).” Ex. 1007 at 71. VDSL1 further explains how to calculate “Erasure Correction” in Table 27 (below) and adds that “the duration of noise pulses the system is protected from could be calculated as $E \times 8/R$, where R is the bit rate of the transmit signal over the line.” *Id.* at 72.

Table 27: Interleaver Characteristics

Parameter	Value	Notes
Block Length (l)	$l = N/8, N/4 \text{ or } N/2$, octets	$N = PS+19$, octets
Depth (D)	$D = M \times l + 1$, octets	$M = 0 \rightarrow 64$, programmable
Erasure Correction (E)	$E = \lfloor t \times l / N \rfloor \times (M \times l + 1)$, octets	$t = 8$ (RS error correction ability)
End-to-End Delay (DL)	$DL = M \times l \times (l - 1)$, octets	
Interleaver Memory Size	$MEM = M \times l \times (l - 1) / 2$, octets	
NOTE 1: The interleaver erasure correction E defines the maximum number of corrupted sequential octets could be corrected by RS algorithm when interleaving is applied. Correspondingly, the duration of noise pulses the system is protected from could be calculated as $E \times 8/R$, where R is the bit rate of the transmit signal over the line.		
NOTE 2: The $\lfloor \rfloor$ symbols indicates the truncation of the value to the lower integer.		

It likewise adds that the “maximum value of [parameter] M required for the given transmission profile shall provide an erasure correction capability up to 500 μ s.” *Id.* at 72.

263. Also, Table B.1 shows how the determining is based on an impulse noise protection requirement.

Table B.1: Interleaving depth Parameter values

Line Rate, Mb/s		1,62	3,24	6,48	12,96	25,92	51,84
Value of N/I		8					
250 μ s of erasure correction	M, octets	2	4	8	16	32	64
	Delay, msec	5,9					
500 μ s of erasure correction	M, octets	4	8	16	32	64	128
	Delay, msec	11,8					

264. Consider Table B.1 for the example of $N=240$, which implies that $I = 30$ since Table B.1 indicates that $N/I = 8$. The table shown below uses the equation $M \times I \times (I-1)/2$ specified by VDSL1 to determine the memory required by the triangular implementation of convolutional interleaving for each set of parameters M and I . Note that the table shows how different amounts of memory are required by VDSL1 for different impulse noise, for different latencies, and for different bit rates.

Impulse noise burst time in ms	0.25 ms			0.5 ms		
Corresponding latency in ms	5.9 ms			11.8 ms		
Bit Rate in Mb/s	M	I (for N=240)	Memory (bytes)	M	I for (N=240)	Memory (bytes)
1.62 Mb/s	2	30	870	4	30	1740
3.24 Mb/s	4	30	1740	8	30	3480
6.48 Mb/s	8	30	3480	16	30	6960
12.96 Mb/s	16	30	6960	32	30	13920
25.92 Mb/s	32	30	13920	64	30	27840
51.84 Mb/s	64	30	27840	128	30	55680

viii. Claim 3: The system of claim 1, wherein the determining is based on a latency requirement.

265. It is my opinion that VDSL1 in view of Fadavi-Ardekani discloses the system of claim 1, wherein the determining is based on a latency requirement . For the system of claim 1, see above, ¶¶ 214–268.

266. VDSL discloses basing the determining of interleaver and deinterleaver memory on a latency requirement. For example, in its explanation of interleaving and deinterleaving, it explains that the “interleaver and de-interleaver shall be adjustable via the management system to meet latency requirements.” Ex. 1007 at 59. It continues by explaining:

The latency of the slow path is a function of the data rate and burst error correction capability. For data rates greater than or equal to 13 Mbps, the latency between the α and β interfaces shall not exceed 10 ms when the interleaver delay is set to the maximum. At lower data rates there is a trade-off between higher latency and decreased burst error correction ability. At any data rate, the minimum latency occurs when the interleaver is turned off.

267. VDSL1 further contemplates various interleaver parameters that affect the determination of the interleaver and deinterleaver memory allocation, including “End-to-end delay,” i.e., latency (below,). Ex. 1007 at 72.

Table 27: Interleaver Characteristics

Parameter	Value	Notes
Block Length (I)	$I = N/8, N/4 \text{ or } N/2$, octets	$N = PS+19$, octets
Depth (D)	$D = M \times I + 1$, octets	$M = 0 \rightarrow 64$, programmable
Erasure Correction (E)	$E = \lfloor t \times I / N \rfloor \times (M \times I + 1)$, octets	$t = 8$ (RS error correction ability)
End-to-End Delay (DL)	$DL = M \times I \times (I - 1)$, octets	
Interleaver Memory Size	$MEM = M \times I \times (I - 1) / 2$, octets	
NOTE 1: The interleaver erasure correction E defines the maximum number of corrupted sequential octets could be corrected by RS algorithm when interleaving is applied. Correspondingly, the duration of noise pulses the system is protected from could be calculated as $E \times 8/R$, where R is the bit rate of the transmit signal over the line.		
NOTE 2: The $\lfloor \rfloor$ symbols indicates the truncation of the value to the lower integer.		

268. Also, Table B.1 shows how the determining is based on a latency requirement.

Table B.1: Interleaving depth Parameter values

Line Rate, Mb/s		1,62	3,24	6,48	12,96	25,92	51,84
Value of N/I		8					
250 μ s of erasure correction	M, octets	2	4	8	16	32	64
	Delay, msec	5,9					
500 μ s of erasure correction	M, octets	4	8	16	32	64	128
	Delay, msec	11,8					

269. Consider Table B.1 for the example of $N=240$, which implies that $I = 30$ since Table B.1 indicates that $N/I = 8$. The table shown below uses the equation $M \times I \times (I-1)/2$ specified by VDSL1 to determine the memory required by the triangular implementation of convolutional interleaving for each set of parameters M and I . Note that the table shows how different amounts of memory are required by VDSL1 for different impulse noise, for different latencies, and for different bit rates.

Impulse noise burst time in ms	0.25 ms			0.5 ms		
Corresponding latency in ms	5.9 ms			11.8 ms		
Bit Rate in Mb/s	M	I (for N=240)	Memory (bytes)	M	I for (N=240)	Memory (bytes)
1.62 Mb/s	2	30	870	4	30	1740
3.24 Mb/s	4	30	1740	8	30	3480
6.48 Mb/s	8	30	3480	16	30	6960
12.96 Mb/s	16	30	6960	32	30	13920
25.92 Mb/s	32	30	13920	64	30	27840
51.84 Mb/s	64	30	27840	128	30	55680

ix. Claim 4: The system of claim 1, wherein the determining is based on a bit error requirement.

270. It is my opinion that VDSL1 in view of Fadavi-Ardekani discloses the system of claim 1, wherein the determining is based on a bit error requirement .

For the system of claim 1, see above, ¶¶ 214–268.

271. VDSL1 discloses the determination of the amount of memory allocated to the interleaver to be based on a bit error requirement. Indeed, VDSL explains that bit error rate and latency are intertwined. Ex. 1007 at 59. Thus, one is left balancing the latency requirements with bit error requirements. Specifically, VDSL1 explains:

The latency of the slow path is a function of the data rate and burst error correction capability. For data rates greater than or equal to 13 Mbps, the latency between the α and β interfaces shall not exceed 10 ms when the interleaver delay is set to the maximum. At lower data rates there is a trade-off between higher latency and decreased burst error correction ability. At any data rate, the minimum latency occurs when the interleaver is turned off.

Id.

XIV. Revision or Supplementation

272. In this report, I have presented my opinions regarding the invalidity of the challenged claims of the '048 Patent based on the information available to me. My opinions are subject to change in view of opinions provided by the Patent Owner or its expert, or any additional information that I may receive. I reserve the right to supplement my opinions accordingly.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Executed October 25, 2022

Richard Wesel

Dr. Richard Wesel