UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AMAZON.COM, INC. AND AMAZON.COM SERVICES LLC,
PETITIONERS

v.

NOKIA TECHNOLOGIES OY,
PATENT OWNER.

IPR2024-00572

U.S. Patent No. 7,724,818

DECLARATION OF DR. ELI SABER, Ph.D. REGARDING U.S. PATENT NO. 7,724,818

TABLE OF CONTENTS

I, Eli Saber, hereby declare as follows:

I. INTRODUCTION

1. I have been retained on behalf of Amazon.com, Inc. and Amazon.com Services LLC ("Amazon" or "Petitioner") to offer technical opinions related to U.S. Patent No. 7,724,818 ("the '818 patent") (EX-1001). I understand that Amazon is requesting that the Patent Trial and Appeal Board ("PTAB" or "Board") institute an inter partes review ("IPR") proceeding of the '818 patent. Specifically, I have been asked to consider the validity of claims 6-9, 11, 15, 21, and 23 of the '818 patent (the "Challenged Claims") in view of prior art and obviousness considerations from the perspective of a person of ordinary skill in the art at the time of the invention ("POSITA") as it relates to the '818 patent.

2. I am being compensated for my time at my standard consulting rate $500/hour ($300/hour for travel time). I am also being reimbursed for expenses that I incur during the course of this work. My compensation is not contingent upon the results of my study, the substance of my opinions, or the outcome of any proceeding involving the Challenged Claims. I have no financial interest in the outcome of this matter or on the pending litigation between Petitioner and Patent Owner.

3. My analysis here is based on my years of education, research and experience, as well as my investigation and study of relevant materials, including those cited herein.

4. I may rely upon these materials, my knowledge and experience, and/or

additional materials to rebut arguments raised by the Patent Owner. Further, I may also consider additional documents and information in forming any necessary opinions, including documents that may not yet have been provided to me.

5. My analysis of the materials in this proceeding is ongoing, and I will continue to review any new material as it is provided. This declaration represents only those opinions I have formed to date. I reserve the right to revise, supplement, and/or amend my opinions stated herein based on new information and on my continuing analysis of the materials already provided.

II. BACKGROUND AND QUALIFICATIONS

6. I am over the age of 18 and am competent to write this declaration. I have personal knowledge, or have developed knowledge of these technologies, based upon education, training, or experience, of the matters set forth herein.

7. My qualifications are set forth in my curriculum vitae, a copy of which is filed as Exhibit EX-1004. The following is a brief summary of my relevant qualifications and professional experience.

8. I am a Professor in the Department of Electrical and Microelectronic Engineering (EME) and the Chester F. Carlson Center for Imaging Science at the Rochester Institute of Technology (RIT) in Rochester, New York. I also serve as the Director of the Image, Video and Computer Vision Laboratory. I joined RIT's full-time faculty in 2004. I have over thirty-four (34) years of industry and academic experience, thirty (30) of which are in the fields of Computer Vision, Digital Image/Video

Processing, Pattern Recognition, and Machine Learning.

9. I received a Ph.D. in Electrical Engineering from the University of Rochester in 1996. My concentration was in Signal/Image/Video Processing, Pattern Recognition, and Computer Vision. I received a Master of Science Degree in Electrical Engineering from the University of Rochester in 1992 in the same concentration as listed above, and I received a Bachelor of Science Degree in Electrical and Computer Engineering, Summa Cum Laude, from the State University of New York at Buffalo in 1988.

10. Before becoming a full-time professor, I worked for Xerox Corporation for 16 years, from 1988 to 2004. During my years at Xerox, I was responsible for delivering color management, image processing innovations, architectures, and algorithms; xerographic sub-systems for a variety of color products; and control systems for toner production facilities. One of several roles I held at Xerox was Advanced Development Scientist and Manager. In that capacity, I established the Advanced Design Laboratory—an Imaging/Xerographics lab—and provided technical and managerial leadership for the Electrical, Imaging and Xerographics Department. In another role, as Product Development Scientist and Manager, I led the research and development of image quality metrics for various product platforms. I also led the Image Science, Analysis, and Evaluation area, with 12 to 15 direct reports and a budget of approximately $2 million.

11. From 1997 until 2004, I was an adjunct faculty member at the Electrical

Engineering Department of the Rochester Institute of Technology and at the Electrical & Computer Engineering Department of the University of Rochester. I was responsible for teaching undergraduate and graduate coursework such as Digital Signal Processing, Digital Image Processing, Digital Video Processing, Pattern Recognition, Detection and Estimation Theory and Communication Systems; and performing research in multimedia applications, pattern recognition, image understanding and color engineering.

12. Since joining RIT full time, I have been responsible for teaching several undergraduate and graduate courses in Digital Signal Processing, Digital Image Processing, Digital Video Processing, Pattern Recognition, Engineering Analysis, Random Signal & Noise, Advanced Engineering Mathematics, Matrix Methods, Communications, Modern Control Theory, and Linear Systems.

13. I proposed, developed and introduced the Digital Video Processing course and reestablished the Pattern Recognition course at the Electrical Engineering Department at RIT. These graduate courses were instrumental in providing MS/PhD students with the proper foundation and knowledge in all aspects of the subject areas in order to perform in-depth research in the field and continue to advance the state of the art.

14. The Digital Video Processing course is designed to provide a thorough and comprehensive coverage of the field. Topics covered include the mathematical representation of digital video including motion and time-varying image formation

models, spatio-temporal sampling/reconstruction, sampling structure conversions, motion estimation using optical flow, block-based, pel-recursive, Bayesian and point correspondence type methodologies, segmentation, motion tracking, filtering, restoration, super-resolution and compression.

15. The Pattern Recognition course, on the other hand, covers pertinent concepts in Bayes decision theory, linear and nonlinear classifiers, artificial neural networks, feature selection/generation/extraction and various methodologies of clustering.

16. I am currently developing a new graduate course titled: “Advanced Neural Networks and Deep Learning: Theory, Mathematics and Algorithms”. The course is intended to provide an in-depth comprehensive study of the theory, mathematics and algorithms for advanced neural networks and their usage in deep learning. It is primarily targeted for upper graduate students that are planning to pursue advanced studies/research in the field. The course provides a thorough study of shallow and deep neural networks architectures and algorithms, activation functions, backpropagation, initialization, optimization strategies, etc. It then progresses to convolutional and recurrent style neural networks and, if time permits, will provide an overview of attention mechanisms/transformers and deep reinforcement learning. The course is being taught for the first time in Spring 2024.

17. My current research focuses on image/video processing for multimedia, military & biomedical applications; computer vision and three-dimensional scene

reconstruction; and machine learning for object tracking/remote sensing applications. As a principal investigator (PI) or co-principal investigator, I have acquired research funding in excess of $5.8 million since joining RIT and have managed multiple government grants from the Department of Defense as well as several corporate grants from Hewlett-Packard, Lenel, and Data Physics. I am currently managing (as PI) a multiyear government grant from the Department of Defense titled "Target Detection/Tracking and Activity Recognition from Multimodal Data." In 2012, I was awarded the Prestigious Trustees Scholarship, the highest award at RIT with regard to research recognition.

18. I am the author or co-author of thirty-eight (38) peer-reviewed journal publications.

19. I have also authored or co-authored one hundred (100) conference and workshop publications and a book entitled Advanced Linear Algebra for Engineers with MATLAB, published by CRC Press in February 2009, and am a named inventor on multiple U.S. and foreign patents.

20. I am a senior member of the Institute of Electrical and Electronic Engineers and a member of the IEEE Signal Processing Society, the Electrical Engineering Honor Society, Eta Kappa Nu.

III. LEVEL OF ORDINARY SKILL IN THE ART

21. In rendering the opinions set forth in this declaration, I was asked to consider the patent claims and the prior art through the eyes of a POSITA at the time of

the alleged invention, which I understand to be April 30, 2003. I understand that the factors considered in determining the ordinary level of skill in a field of art include the level of education and experience of persons working in the field; the types of problems encountered in the field; the teachings of the prior art; and the sophistication of the technology at the time of the alleged invention. I understand that a POSITA is not a specific real individual, but rather is a hypothetical individual having the qualities reflected by the factors above. I understand that a POSITA would also have knowledge from the teachings of the prior art, including the art cited below.

22. Taking these factors into consideration, in my opinion, at the relevant time, a POSITA relating to the technology of the ’818 patent would have had a (1) Bachelor’s degree in electrical engineering, computer engineering, computer science, or a comparable field of study such as physics, and (2) approximately two to three years of practical experience with video encoding/decoding. In my opinion, additional experience could substitute for the level of education and vice-versa.

23. Before April 30, 2003, my level of skill in the art was at least that of a POSITA. I am qualified to provide opinions concerning what a POSITA would have known and understood at that time, and my analysis and conclusions herein are from the perspective of a POSITA as of that date.

IV. MATERIALS CONSIDERED AND RELIED UPON

24. In forming my opinions expressed herein, in addition to relying on my education and experience, I considered the totality of the following materials:

Exhibit	Reference
1001	U.S. Patent No. 7,724,818 (“the ’818 patent” or “’818Pat.”)
1002	File History of the ’818 patent (“’818FH”)
1005	U.S. Patent No. 7,660,356 to Linzer (“Linzer”)
1006	Certified File History of Linzer (“LinzerFH”)
1007	Editor's Proposed Draft Text Modifications for Joint Video Specification (IUT-T Rec. H.264 ISO/IEC 14496-10 AVC), Draft 7 (“JVT-E022d7”) (excerpted from LinzerFH)
1008	U.S. Patent No. 6,646,578 (“Au”)
1009	Nokia Technologies Oy v. Amazon.com, Inc. et al., Case No. 1:23-cv-01236, ECF No. 6, (D. Del. Nov. 3, 2023) (Summons served 10/31/2023).
1010	Nokia Corporation et al. v. Amazon.com, Inc. et al., Case No. 1:23-cv-01232, ECF No. 8, (D. Del. Nov. 3, 2023) (Summons served 10/30/2023).
1011	February 2, 2024 Email from Joshua Newcomer with subject line “1380: Nokia’s proposed constructions.”
1012	H.26L over IP and H.324 Framework (“VCEG-N52”), available at https://www.itu.int/wftp3/av-arch/video-site/0109_San/VCEG-N52.doc.
1013	Analysis of fractional pixel interpolation for MC prediction (“VCEG-N55”), available at https://www.itu.int/wftp3/av-arch/video-site/0109_San/VCEG-N55.doc.
1014	Declaration of June Munford (“Munford Decl.”)
1015	Tekalp, A. (1995). Digital Video Processing.
1016	Certain Video Capable Electronic Devices, Including Computers, Streaming Devices, Televisions, and Components and Modules Thereof, Inv. No. 337-TA-1380 (USITC Nov. 30. 2023), Exhibit 6A to Complaint.
1017	Certain Video Capable Electronic Devices, Including Computers, Streaming Devices, Televisions, and Components and Modules Thereof, Inv. No. 337-TA-1380 (USITC Nov. 30. 2023), Exhibit 6B to Complaint.

Exhibit	Reference
1018	U.S. Patent Publication No. 2004/0161033 ("Notoya")
1019	William Stallings, Computer Organization and Architecture, Designing for Performance, 8th Edition (2010) ("Stallings").
1020	U.S. Patent No. 7,227,901 ("Joch")
1021	U.S. Patent No. 6,529,059 ("Sriram")
1022	Video coding for low bit rate communication (ITU-T H.263 Series H: Audiovisual and Multimedia Systems, Infrastructure of audiovisual services – Coding of moving video)
1023	John Watkinson, The MPEG Handbook, Second Edition (2004)
1024	Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, Second Edition (1988)
1025	Declaration of Dr. Clifford Reader ("Reader")

I have also considered any materials cited in this declaration even if not included in the list above.

V. LEGAL STANDARDS

25. I am not a lawyer and do not provide any legal opinions, but I have been advised that certain legal standards are to be applied by technical experts in forming opinions regarding meaning and validity of patent claims. I have applied the legal standards described below.

26. It is my understanding that assessing the validity of a U.S. patent based on a prior art analysis involves two steps. First, one must assess what meaning a POSITA would have given the terms. Second, after assessing the meaning of the claim terms, one may then assess validity by comparing a patent claim to the "prior art." I understand that the teaching of the prior art is viewed through the eyes of a POSITA at the time the

invention was made. My analysis as to what constitutes a relevant POSITA is set forth above.

A. Legal Standards for Priority Date

28. I understand that the "priority date" (or "earliest effective filing date") of a patent is the date on which it is filed, or the date on which an earlier application was filed if the patentee properly claims the benefit of the earlier application's filing date. For purposes of my analysis in this declaration, I have assumed that the '818 patent is entitled to a priority date of April 30, 2003.

B. Legal Standards for Obviousness

29. My understanding is that a patent claim is invalid as obvious only if the subject matter of the claimed invention "as a whole" would have been obvious to a POSITA at the time the invention was made. To determine the differences between a prior art reference (or a proposed combination of prior art references) and the claims, the question of obviousness is not whether the differences themselves would have been obvious, but whether the claimed invention as a whole would have been obvious. Also, obviousness grounds cannot be sustained by mere conclusory statements. Rather, it is necessary to provide some articulated reasoning with rational underpinning to support the legal conclusion of obviousness.

30. I understand that a patent claim that comprises several elements is not proved obvious by simply showing that each of its elements was independently known in the prior art. In my evaluation of whether any claim of the '818 patent would have

been obvious, I considered whether a POSITA would have had a reason with a rational basis that would have motivated a POSITA to combine the elements or concepts from the prior art in the same way as in the claimed invention.

31. It is my understanding that there is no single way to define the line between true inventiveness on one hand—which is patentable—and the application of common sense and ordinary skill to solve a problem on the other hand—which is not patentable. For instance, factors such as market forces or other design incentives may be the source of what produced a change, rather than true inventiveness.

32. I understand that the decision-maker may consider whether the change was merely the predictable result of using prior art elements according to their known functions, or whether it was the result of true inventiveness. And the decision-maker may also consider whether there is some teaching or suggestion in the prior art to make the modification or combination of elements recited in the claim at issue. Also, the decision-maker may consider whether the innovation applies a known technique that had been used to improve a similar device or method in a similar way. The decision-maker may also consider whether the claimed invention would have been obvious to try, meaning that the claimed innovation was one of a relatively small number of possible approaches to the problem with a reasonable expectation of success by those skilled in the art.

33. I understand that in considering obviousness, it is important to be careful not to engage in hindsight and that many true inventions might seem obvious after the

fact. I understand that obviousness is considered from the perspective of a POSITA at the time the claimed invention was made, and not based on what is known today or what is learned from the teaching of the patent.

34. I understand that the following factors must be evaluated to determine whether a claim is obvious: the scope and content of the prior art; the difference or differences, if any, between the claim of the patent and the prior art; the level of ordinary skill in the art at the time the claimed invention was made; and, if available, the objective indicia of non-obviousness, also known as "secondary considerations."

35. I understand that the secondary considerations can include: commercial success of a product due to the merits of the claimed invention; a long felt need for the solution provided by the claimed invention; unsuccessful attempts by others to find the solution provided by the claimed invention; copying of the claimed invention by others; unexpected and superior results from the claimed invention; acceptance by others of the claimed invention as shown by praise from others in the field or from the licensing of the claimed invention; teaching away from the conventional wisdom in the art at the time of the invention; independent invention of the claimed invention by others before or at about the same time as the named inventor thought of it; and other evidence tending to show obviousness.

36. I understand that, to establish a secondary consideration, the evidence must demonstrate a nexus between that secondary consideration and the claimed invention.

37. I understand that a petition must identify, for any terms with means-plus-

function limitations, the specific portions of the specification that describe the structure, material, or acts corresponding to each claimed function. I also understand a rebuttable presumption exists that with use of the term, “means,” a claim term is interpreted as means-plus-function.

38. I also understand that a petitioner seeking to demonstrate that the prior art discloses a means-plus-function limitation must prove that the corresponding structure—or an equivalent—was present in the prior art and performing the same claimed function.

39. I understand that the standard for determining whether the prior art discloses the corresponding structure or an equivalent is whether there are insubstantial differences, which means that if the assertedly equivalent structure performs the claimed function in substantially the same way to achieve substantially the same result as the corresponding structure described in the specification, it is equivalent.

40. I understand that proving structural equivalence does not require a component-by-component analysis, but rather that it is sufficient to show structure, materials, or acts that perform the claimed function in substantially the same way to achieve substantially the same result because the claim limitation is the overall structure corresponding to the claimed function and not individual components.

VI. SUMMARY OF OPINIONS

41. For the reasons I discuss below, it is my opinion that claims 6-9, 11, 15, 21, and 23 of the ’818 patent are disclosed in the prior art, and are rendered obvious by

the prior art.

VII. OVERVIEW OF THE TECHNOLOGY AND THE ’818 PATENT

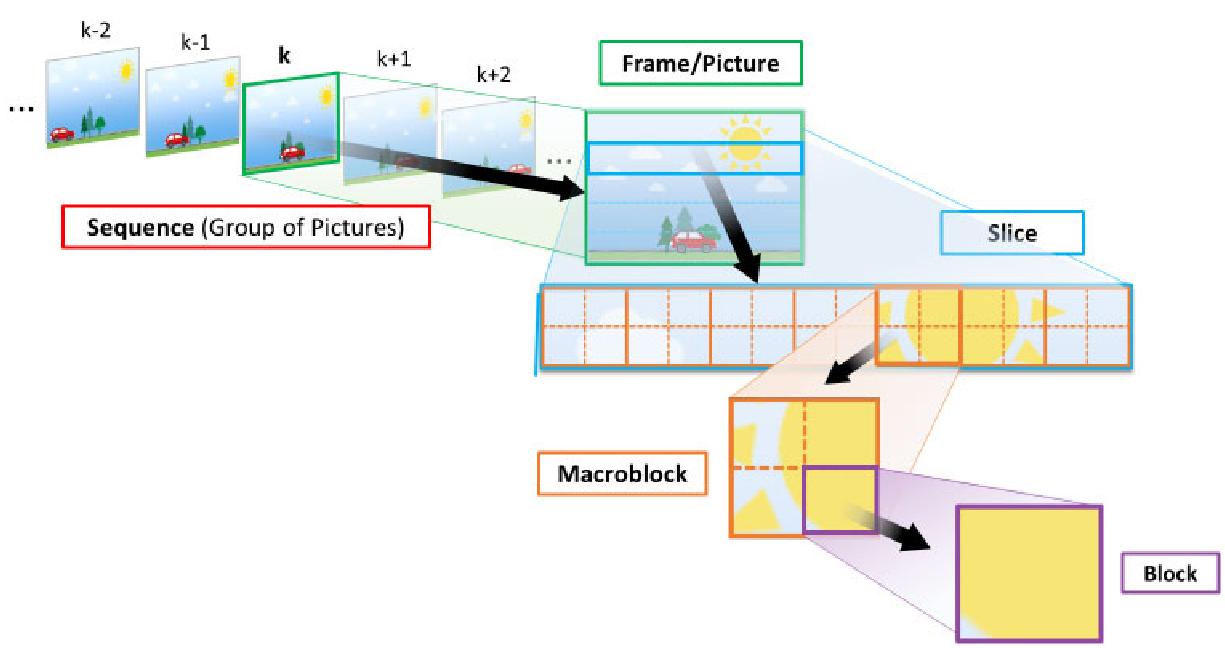
42. The ’818 patent is titled “Method for coding sequences of pictures” and issued on May 25, 2010, from Application 10/426,928 (“’928 application”), filed on April 30, 2003.

A. Background of Technology and Admitted Prior Art

43. The ’818 patent relates to “[a] method for encoding [and decoding] sequences of pictures into [and from] a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices.” EX-1001, Abstract, 1:6–12. The ’818 patent aims to “provide a more efficient method to transmit parameters relating to picture information.” *Id.*, 3:55–58. Digital video involves sending and receiving video data in compressed form. EX-1015, 6. This includes processing for motion analysis, standards conversion, enhancement, and restoration. *Id.*, 7. Compression is fundamental to digital video applications. *Id.* In addition, to process digital video, encoders and decoders have been generally used and well known in the art since at least 1995. For example, A. Murat Tekalp, Digital Video Processing, Prentice Hall 1995 (“Tekalp” EX-1015), describes the “MPEG-1 Encoder and Decoder,” noting that an “MPEG-1 decoder reverses the operations of the encoder.” *Id.*, 11–12.

44. Going at least back to the MPEG-1 standard, adopted in 1992, digital video has followed a hierarchy, where a sequence comprises a group of pictures (GOP), a

GOP comprises pictures, each picture comprises slices, slices comprise macroblocks, and macroblocks comprise blocks. *Id.*, 10. A picture can be either a frame or a field (e.g., a frame consists of a top field and a bottom field). *Id.*, 13. Fields are used typically in interlaced video. *Id.*; *see also* John Watkinson, The MPEG Handbook, ("Watkinson" EX-1023), 9 (a "picture" "could be a field or a frame in interlaced systems but only a frame in non-interlaced systems"). Because most modern digital video uses progressive scans (and not interlaced), frames are synonymous with pictures in most applications. The image shown below illustrates the relationship between blocks, macroblocks, slices, frames/pictures, and a GOP.[1]



45. The "Background of the Invention" section of the '818 patent admits that various video coding standards, as well as VCEG-N52, were known. EX-1001, 1:18–

[1] This image was created under my direction for the purposes of this declaration.

30. In addition, the ’818 patent describes the “parameter set concept,” which it acknowledges was described in VCEG-N52[2], and describes known aspects of that concept. *Id.*, 2:48–3:28; *see also* EX-1012, 8. According to the ’818 patent, VCEG-N52 included a parameter set with “all picture, GOP, and sequence level data” (e.g., picture size, display window, optional coding modes employed, macroblock allocation map) and a unique identifier. EX-1001, 2:53–57. Further, in VCEG-N52, “[e]ach slice header includes a reference to a parameter set identifier, and the parameter values of the referred parameter set are used when decoding the slice.” *Id.*, 2:57–58. Importantly, the ’818 patent acknowledges VCEG-N52 could have multiple parameter sets, transmitting a codeword indicating which parameter set to be used in decoding. *Id.*, 3:5–10.

46. The ’818 patent discloses a conventional encoder and decoder within which the VCEG-N52’s known parameter set concept could be implemented. EX-1001, 3:55–4:1. As described above, in my personal experience, encoders and decoders were generally well known at the relevant time. For example, Tekalp describes the “MPEG-1 Encoder and Decoder,” noting that an “MPEG-1 decoder reverses the operations of the encoder.” EX-1015, 11–12. Further, in the ’818 patent, Figure 2

[2] Although the ’818 patent refers to “VCEG-N55,” the reference should have been to document “VCEG-N5**2.**” I base this understanding on (1) the References Cited section of the ’818 patent, which includes “H.26L over IP and H.324 Framework” (a title of the VCEG-N52 document) (EX-1001, References Cited; EX-1012), and (2) the applicant’s reference to VCEG-N52 in response to a rejection based on Visharam (’818FH, 146), and (3) that VCEG-N55 is not related to parameters or parameter sets (*see* EX-1013).

depicts only conventional elements of an encoder/decoder system, and Figure 4 depicts only conventional decoder components. Specifically, Figure 2 depicts encoder 1 and decoder 2, and Figure 4 depicts the decoder including processor 2.2 and memory 2.3. In my opinion, a POSITA would have understood that VCEG-N52's parameter set concept discussed in the '818 Background section was implemented on a conventional encoder/decoder system such as that shown in Figures 2 and 4.

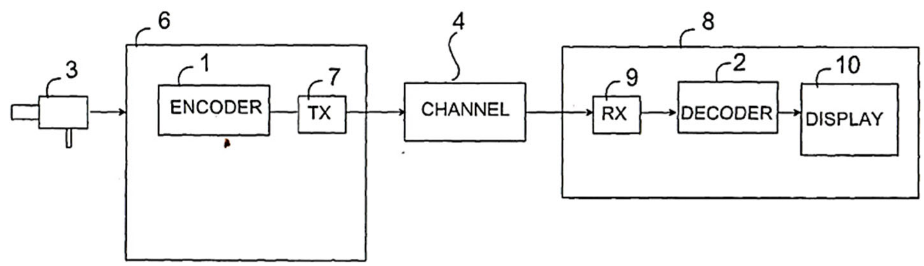


Fig. 2

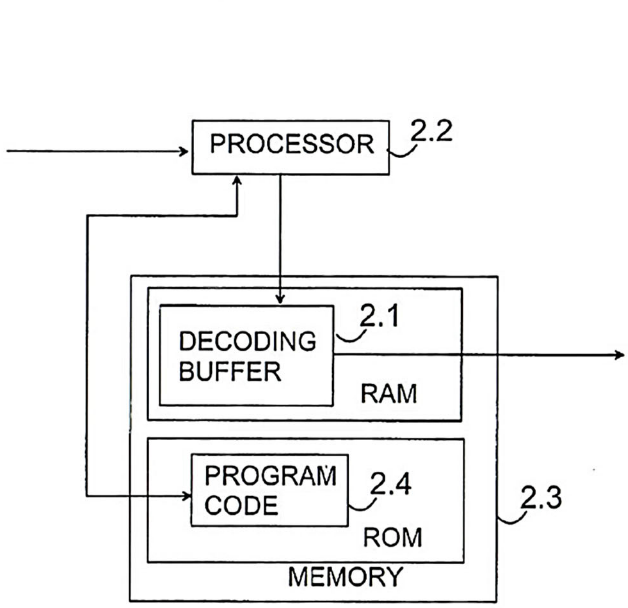


Fig. 4

B. Purported Invention

47. The purported invention of the ’818 patent splits the admittedly-known parameter sets into “multiple parameter set structures according to the persistency and target of parameters.” EX-1001, 4:15–18. The parameter sets include “a sequence parameter set,” and “a picture parameter set,” as well as picture parameters included in a “slice header.” *Id.*, Abstract, 4:41–48. The image below shows how the parameter sets, illustrated in Figure 1 of the ’818 patent, are related to a coded video sequence, an example of which is shown above. As shown in the figure below, Figure 1’s parameter sets include independent GOP parameter set 105 (annotated red), picture parameter set 104 (annotated green), and slice parameter set 103 (annotated blue).

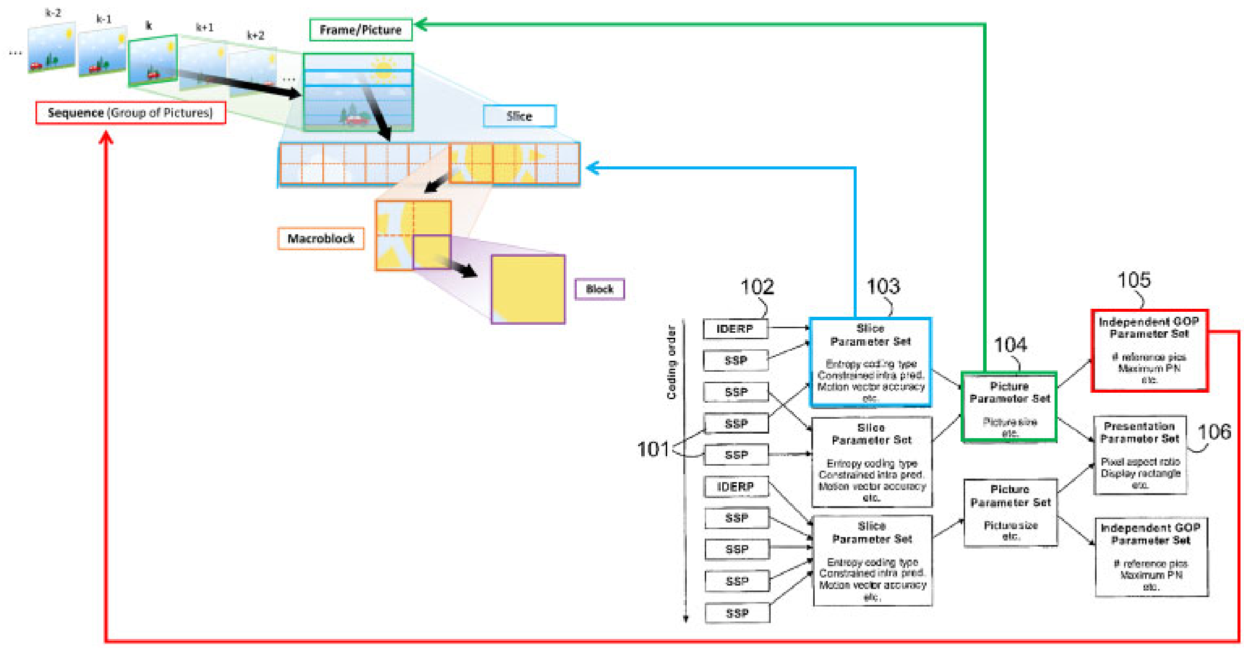


Fig. 1

48. According to the '818 patent, certain parameter values must remain unchanged throughout a GOP or within a picture. *Id.*, 7:25–30. Parameter values that likely remain unchanged in multiple pictures but are allowed to change in every picture are included in the picture parameter set. *Id.*, 4:17–30. Parameter values that are not allowed to change in multiple pictures are included in the sequence parameter set. *Id.* Parameter values that may change in every slice or in every picture, are included in the slice header. *Id.* However, the '818 patent's method of decoding includes defining at least one picture parameter in a slice header that remains unchanged in all slice headers of one picture. *Id.*, 4:50–62. According to the '818 patent, by assigning parameters this way "it is clear that all picture parameter sets that are referred to within a sequence must refer to the same sequence parameter set. Moreover, it is clear that all slices of a picture must refer to the same picture parameter set." *Id.*, 7:1–5.

C. Brief Summary of ’818 Patent Prosecution History

49. I have reviewed the file history that led to the issuance of the ’818 patent.

50. It is my understanding, based on my review of the prosecution history, that the application leading to the ’818 patent underwent years of examination prior to issuance, and the Examiner repeatedly found that the prior art taught the purported invention in that application (e.g., the “separation of parameter sets to more than one category, including both a sequence parameter set and picture parameter set”) as discussed in the ’818 specification and specifically argued by the applicant during prosecution. E.g., ’818FH, 111, 142, 164, 166–67, 334–35, 363–64, 395–96.

51. Similar to issued claim 6, originally filed independent claim 6 recited “recognizing a sequence parameter set and forming at least one sequence parameter pertaining to a sequence using the parameter set; recognizing a picture parameter set and forming at least one first picture parameter pertaining to a picture using the parameter set; and forming at least one second picture parameter using information of a slice header, the at least one second picture parameter remaining unchanged at least in all slice headers of one picture; and using the at least one second picture parameter in decoding.” ’818FH, 31. Similar features were recited in other originally-filed independent claims.

52. During prosecution, the examiner issued five office actions, finding first that the claims were anticipated by U.S. Patent Application No. 2004/0010802 (“Visharam”) and later that the independent claims were unpatentable as obvious over

Visharam in view of U.S. Patent No. 7,227,901 ("Joch"). *Id.*, 111, 164, 334, 363, 395–96. The examiner specifically found Visharam disclosed a "sequence parameter set" and "picture parameter set," even considering applicant's argument that its invention was a "separation of parameter sets to more than one category, including both a sequence parameter set and picture parameter set." *E.g., id.* at 142–43, 165, 178–87, 334, 351. The Examiner later relied on the combination of Visharam and Joch, and specifically on Joch for the picture parameter remaining unchanged in all slice headers of a picture. *Id.*, 363–64; *see also id.*, 381, 392, 395–96. During prosecution, the applicant argued that Visharam disclosed "no more than what is included in" VCEG-N52, and provided VCEG-N52 to the Examiner. *Id.*, 142.

53. Over six years after the application was filed, the applicant amended the pending independent claims corresponding to those challenged here to recite, *inter alia*, forming "picture parameter values" instead of forming "picture parameters," including that the "second picture parameter value" (rather than second picture parameter) "remain[s] unchanged at least in all slice headers of one picture." *Id.*, 403–04, 406–09. The applicant argued that the references did not teach "defining at least one picture parameter value in a slice header, the picture parameter value remaining unchanged at least in all slice headers of one picture." *Id.*, 404.

54. Following that, the examiner issued a Notice of Allowance for all pending claims, without specifying any reasons for allowance. *Id.*, 417–20.

VIII. CLAIM CONSTRUCTION

55. I have given all the claim terms of the Challenged Claims their plain and ordinary meaning, as would have been understood by a POSITA, at the time of the alleged invention, unless otherwise discussed below.

A. Means-Plus-Function (MPF)

56. It is my understanding that when a claim limitation uses the term "means" and includes functional language, it is presumed to be a means-plus-function term. It is further my understanding that presumption is overcome only if the limitation further includes the structure, material, or acts necessary to perform the recited function.

57. Claim 23 recites limitations as "means for" performing a function. I understand that Petitioner identified the structure below as the structure corresponding to the claimed functions (based on PO's allegations in the ITC investigation). *See* EX-1011. I have reviewed the '818 specification and agree that this structure performs the claimed functions in the specification. I do not offer an opinion on whether or not that structure is proper or sufficient for legal purposes under MPF.

Limitation (Function Emphasized)	Corresponding Structure
"means for *recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set for a*	"a decoder as described in the '818 patent specification in figure 4 and equivalents thereof."

picture”	
“means for *recognizing a picture parameter set and forming at least one first picture parameter value pertaining to a picture using the parameter set*”	“a decoder as described in the ’818 patent specification in figure 4 and equivalents thereof.”
“means for *forming at least one second picture parameter value using information of a slice header*, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture”	“a decoder as described in the ’818 patent specification in figure 4 and equivalents thereof.”
“means for *using the at least one second picture parameter value in decoding*”	“a decoder as described in the ’818 patent specification in figure 4 and equivalents thereof.”

IX. DETAILED EXPLANATION OF GROUNDS

A. Ground 1: Obviousness over JVT-E022d7 (Claims 6–9, 11, 15, 21, and 23)

58. It is my opinion that claims 6–9, 11, 15, 21, and 23 of the ’818 patent would have been obvious in view of JVT-E022d7, either on its own or in combination with the knowledge of a POSITA.

1. JVT-E022d7

59. It is my understanding that JVT-E022d7 was a printed publication, which was publicly available before earliest priority date on the face of the ’818 patent. Munford, ¶¶4–17; Reader, ¶¶28–50. This understanding is based on JVT-E022d7 itself, which indicates it was “Generated: 2002-09-19,” for an October 2002 meeting in Geneva, Switzerland. I also understand that June Munford and Clifford Reader each provided declarations (EX-1014 and EX-1025, respectively) that described the public availability of this document. Therefore, it is my understanding that JVT-E022d7 is prior art under §102(a) (pre-AIA).

60. As indicated by its title, JVT-E022d7 was an editor’s proposed draft of what would become the H.264 video standard. JVT-E022d7, 1.[3] JVT-E022d7 discloses “an industry standard for compressed video representation with substantially increased coding efficiency and enhanced robustness to network environments.” *Id.*, 14. According to JVT-E022d7, a “video source that is represented by the bitstream is a sequence of frames and/or fields (called collectively pictures) in decoding order.” *Id*, 27. Further, a “picture” is a “field or a frame” where a “frame contains sampled and quantized luma and chroma data of all rows of a video signal frame.” *Id.*, 20–21. JVT-E022d7 also defined a slice group as “a sub-set of the macroblocks of a picture.” *Id.*, 23. A macroblock, in turn, is “a 16x16 block of luma samples and a two corresponding

[3] Citations to page numbers refer to the page number of the exhibit, which may not match the internal page numbering of the document.

blocks of chroma samples." *Id.*, 16. A POSITA would therefore have understood that JVT-E022d7 discloses that all slices are part of a picture. This description by JVT-E022d7 corresponds to the background disclosure in '818 patent, as explained in Section VII.A above (i.e., bitstream includes sequences of pictures; a picture sequence corresponds to group of pictures; a picture is a frame of a field; a picture consists of one or more slices). Both JVT-E022d7 and the '818 patent thus disclose that all slices are part of a picture.

61. In my opinion, JVT-E022d7 also describes what was part of the purported invention of the '818 patent: splitting the parameters sets into the "sequence parameter set" and the "picture parameter set," as well as picture parameters values included in a "slice header." JVT-E022d7 discloses the use of "sequence parameter set[s]," which contain values that are "replaced only when an IDR [instantaneous decoder refresh] NAL [network abstraction layer] unit refers to a different sequence parameter set." *Id.*, 52. In other words, like the '818 patent's purported invention, each sequence parameter set of JVT-E022d7 is used for a sequence of pictures. And like the '818 patent's purported invention, JVT-E022d7 also discloses the use of "picture parameter set[s]," which "include[] the parameters that remain unchanged within a coded picture." *Id.* Also like the '818 patent, JVT-E022d7 discloses a "slice header" that is "part of a coded slice containing the coded representation of data elements pertaining to the slice data that follow the slice header." *Id.*, 23. Each slice header includes a parameter, "frame_num," which identifies the frame for which the slice is included and the value

of which would not change for all slice headers in a picture. *Id.*, 57.

2. Claim 6

a. 6[pre]: A method for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, the method comprising:

62. If limiting, in my opinion, JVT-E022d7 discloses or suggests the preamble.

63. JVT-E022d7 describes "an industry standard for compressed video representation" which includes a "decoding process" (i.e., a "method for decoding"). *Id.*, 14, 16. As to "decoding sequences of pictures from a bitstream," JVT-E022d7 defines "bitstream" as a "sequence of bits that forms the representation of data and coded fields and frames." *Id.*, 19. JVT-E022d7 discloses "requirements for the [network abstraction layer] NAL unit deliver and decoding order" and that "[d]ecoders conforming to this Recommendation | International Standard shall be capable of receiving NAL units in decoding order." *Id.*, 70. JVT-E022d7 further defines "decoding order" as "[t]he order in which the coded pictures are to be decoded," and states "[t]he decoding order is specified such that the decoding of pictures that use interpicture prediction follows later in decoding order than other pictures that are referenced in the decoding process." *Id.*, 15, 20. In other words, JVT-E022d7 describes decoders that receive NAL units, which are independently decodable components of a bitstream, in a decoding order, where the bitstream includes a sequence of bits forming fields and frames. In my opinion and based on the disclosures I discuss above, a POSITA would have understood the decoding order to be the order in which pictures

from the bitstream are decoded.

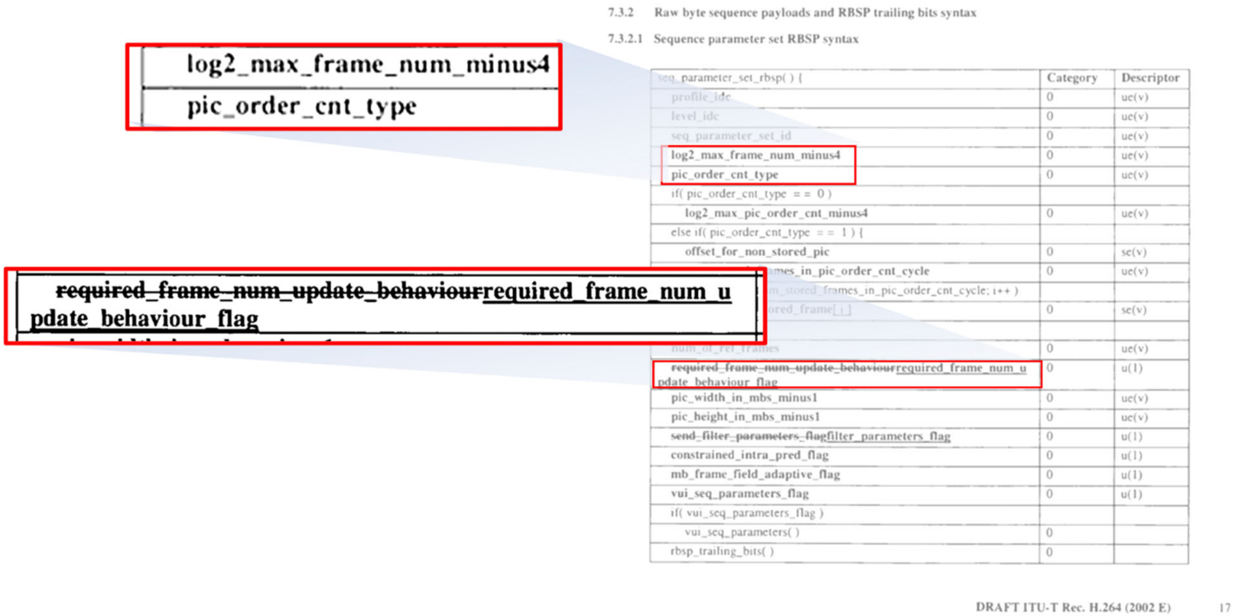
64. As to "parameters are defined in a parameter set," in JVT-E022d7, "Sequence Parameter Set" and "Picture Parameter Set" are types of elements contained in the NAL units. *Id.*, 50. Regarding the sequence parameter set, JVT-E022d7 discloses that the "decoding order of a sequence parameter set shall precede the decoding order of other NAL units that refer to that sequence parameter set." *Id.*, 70. Regarding the picture parameter set, JVT-E022d7 discloses that a "picture parameter set includes the parameters that remain unchanged within a coded picture" and that the "decoding order of a picture parameter set shall precede the decoding order of other NAL units that refer to that picture parameter set." *Id.*, 52, 70. Based on this disclosure, a POSITA would have understood that the sequence parameter set and the picture parameter set have parameters defined in one or more parameter sets. In my opinion, a POSITA would have further understood that the parameter sets are used in decoding of picture sequences, because JVT-E022d7 describes the decoding order for these parameters sets.

65. Further, in JVT-E022d7, "each picture comprises information of one or more slices." Specifically, JVT-E022d7 explains "pictures are divided into macroblocks" and a slice contains an "integer number of macroblocks ordered contiguously in raster scan order within a particular slice group"; each picture, therefore, includes one or more slices, where each slice includes macroblocks. *Id.*, 23, 29. JVT-E022d7 also discloses that "[d]ecoding of a new picture is started from the slice to be decoded." *Id.*, 71. Based on this, in my opinion, a POSITA would have

understood that each picture comprises information of one or more slices, and decoding a picture includes decoding a slice of the picture.

b. 6[a]: recognizing, in a decoder, a sequence parameter set and forming at least one sequence parameter pertaining to a sequence using the parameter set;

66. In my opinion, JVT-E022d7 discloses or suggests 6[a]. JVT-E022d7 discloses the use of a "sequence parameter set," which contain values that are "replaced only when an IDR [instantaneous decoder refresh] NAL [network abstraction layer] unit refers to a different sequence parameter set." *Id.*, 52. JVT-E022d7 also provides examples of parameters for a sequence parameter set in raw byte sequence payloads (RBSP) syntax in Table 7.3.2.1. The sequence parameter set includes parameters that provide information about properties and characteristics of a video sequence, such as picture order ("pic_order_cnt_type), number of reference frames ("num_of_ref_frames"), picture size (e.g., "pic_width_in mbs_minus1" and "pic_height_in mbs_minus1,"). *Id.*, 35. In my opinion, a POSITA would have understood that JVT-E022d7 discloses using a sequence parameter set in decoding because the sequence parameter set includes values needed in decoding of sequences, and such values are specific for a sequence parameter set.



7.3.2 Raw byte sequence payloads and RBSP trailing bits syntax

7.3.2.1 Sequence parameter set RBSP syntax

seq_parameter_set_rbsp( ) {	Category	Descriptor
profile_idc	0	ue(v)
level_idc	0	ue(v)
seq_parameter_set_id	0	ue(v)
log2_max_frame_num_minus4	0	ue(v)
pic_order_cnt_type	0	ue(v)
if(pic_order_cnt_type = = 0)		
log2_max_pic_order_cnt_minus4	0	ue(v)
else if(pic_order_cnt_type = = 1) {		
offset_for_non_stored_pic	0	se(v)
[illegible]mes_in_pic_order_cnt_cycle	0	ue(v)
[illegible]m_stored_frames_in_pic_order_cnt_cycle; i++)		
[illegible]ored_frame[i]	0	se(v)
num_of_ref_frames	0	ue(v)
~~required_frame_num_update_behaviour~~required_frame_num_update_behaviour_flag	0	u(1)
pic_width_in_mbs_minus1	0	ue(v)
pic_height_in_mbs_minus1	0	ue(v)
~~send_filter_parameters_flag~~filter_parameters_flag	0	u(1)
constrained_intra_pred_flag	0	u(1)
mb_frame_field_adaptive_flag	0	u(1)
vui_seq_parameters_flag	0	u(1)
if(vui_seq_parameters_flag)		
vui_seq_parameters( )	0	
rbsp_trailing_bits( )	0	

DRAFT ITU-T Rec. H.264 (2002 E) 17

Id., 35 (annotated).

67. The sequence parameter syntax in Table 7.3.2.1 includes a parameter "Log2_max_frame_num_minus4," (annotated with red) which "specifies the MAX_FN [maximum frame number] used in frame number related arithmetic." *Id.*, 52. "Log2_max_frame_num_minus4" is a parameter of the sequence parameter set that determines a range of frame numbers that can be used in a video sequence.

68. The sequence parameter syntax in Table 7.3.2.1 also includes a parameter "Pic_order_cnt_type" (annotated with red). *Id.*, 35. JVT-E022d7 discloses that "[i]f pic_order_cnt_type is 0, the decoder shall maintain a picture order count offset, called PicOrderCntOffset, which is a 32-bit signed integer [and] [i]f pic_order_cnt_type is 1, the decoder shall maintain a FrameNumOffset that is a 32-bit unsigned integer." *Id.*, 71–72. In my opinion, a POSITA would have understood that the parameter "pic_order_cnt_type" is used for displaying pictures in a picture sequence in a correct

order because the "pic_order_cnt_type" is particularly needed when pictures in a sequence are reordered.

69. The sequence parameter syntax in Table 7.3.2.1 also includes a parameter "required_frame_num_update_behaviour_flag" (annotated with red). *Id.*, 35. Further, JVT-E022d7 discloses that the parameter "required_frame_num_update_behaviour_flag" "equal to 1 specifies a specific decoder behaviour in case of missing frame numbers" and is used in instances where "the decoder identifies that pictures that should have been stored have not been decoded, by a gap in frame numbers, the decoder shall act as if the missing pictures had been inserted into the reference picture buffer using the 'Sliding Window' buffering mode." *Id.*, 53, 80. U.S. Patent Publication No. 2004/0161033 ("Notoya") further describes the understanding of a POSITA regarding the "required_frame_num_update_behaviour_flag" parameter. EX-1018. Notoya states that "[t]he JVT defines the operation of the memory management that when a part of the sequence of pictures inputted by the moving picture decoding apparatus is lost, invalid pictures are inserted for the number corresponding to the lost pictures. This operation is performed by the moving picture decoding apparatus when 'required_frame_num update_behaviour_flag' included in the sequence parameter set indicates '1'." *Id.*, [0035]. Based on these disclosures and in my opinion, a POSITA would have understood that the parameter "required_frame_num_update_behaviour_flag" is used by the decoder to specify

behavior when frame numbers are missing.

70. In my opinion, a POSITA would have understood that "recognizing, in a decoder, a sequence parameter set" as recited in 6[a] encompasses a decoder parsing the incoming bitstream and identifying a "sequence parameter set." JVT-E022d7 performs such "recognizing" because, depending on the implementation, its decoder extracts data elements (such as the sequence parameter set) from the bitstream based on the syntax described in the document or the sequence parameter set is *a priori* predefined in the decoder and the decoder recognizes a reference to the sequence parameter set from the bitstream. JVT-E022d7, 14–15. JVT-E022d7 also discloses copying the decoded parameter sets into locations in the decoder. JVT-E022d7, 53. I have reviewed Exhibits 6A and 6B to the complaint in *Certain Video Capable Electronic Devices, Including Computers, Streaming Devices, Televisions, and Components and Modules Thereof*, Inv. No. 337-TA-1380 (USITC Nov. 30. 2023), in which Patent Owner is the Complainant. EX-1016, EX-1017. It is my understanding that those exhibits chart Patent Owner's allegations regarding how the limitations of the '818 patent are met by the accused products. While Patent Owner provides little explanation for how the "recognizing" term is met, from my review, my understanding of that term is consistent with how Patent Owner appears to understand that term for infringement purposes. A POSITA would have further understood that this parsing, extraction, and copying process also results in "forming" e.g., creating in local memory) the data element, such as, for this limitation "at least one sequence parameter pertaining

to a sequence using the parameter set." Again, while Patent Owner provides little explanation for how the "forming" term is met, from my review discussed above, my understanding of that term is consistent with how Patent Owner appears to understand that term for infringement purposes. A POSITA would have understood JVT-E022d7 teaches the "forming" limitation for the same reasons discussed above, including that it decodes pictures using, for example, a maximum number of reference frames, a picture order count, and a flag indicating missing frame numbers.

c. 6[b]: recognizing, in the decoder, a picture parameter set and forming at least one first picture parameter value pertaining to a picture using the parameter set;

71. In my opinion, JVT-E022d7 discloses or suggests 6[b]. First, JVT-E022d7 discloses the use of a "picture parameter set," which "includes the parameters that remain unchanged within a coded picture." JVT-E022d7, 52. JVT-E022d7 also discloses that "[e]very picture parameter set shall refer to the active sequence parameter set [and that] [t]he picture parameters of an active picture parameter set shall be replaced only when the first slice NAL unit or DPA [Data Partition Type A] NAL unit of a picture refers to a different picture parameter set." *Id.* JVT-E022d7 further discloses (as part of NAL decoding), that the "decoding order of a picture parameter set shall precede the decoding order of other NAL units that refer to that picture parameter set." *Id.*, 70. In my opinion, a POSITA would have understood that JVT-E022d7 discloses using a picture parameter set in decoding, because JVT-E022d7 discloses decoding such parameter sets before decoding other NAL units that refer to that parameter set. JVT-

E022d7 provides examples of parameters for its picture parameter set in RBSP syntax in Table 7.3.2.2.

num_slice_groups_minus1

num_ref_idx_l0_active_minus1

weighted_pred_flag

7.3.2.2 Picture parameter set RBSP syntax

pic_parameter_set_rbsp( ) {	Category	Descriptor
pic_parameter_set_id	1	ue(v)
seq_parameter_set_id	1	ue(v)
entropy_coding_mode	1	ue(v)
motion_resolution	1	ue(v)
adaptive_block_size_transform_flag	1	u(1)
num_slice_groups_minus1	1	ue(v)
if(num_slice_groups_minus1 > 0) {		
mb_allocation_map_type	1	ue(v)
if(mb_allocation_map_type = = 0)		
for(i = 0; i <= num_slice_groups_minus1; i++)		
run_length	1	ue(v)
else if(mb_allocation_map_type = = 2)		
for(i = 0; i < num_mbs_in_pic; i++)		
slice_group_id [Ed. Note: how many bits ?]	1	u(v)
else if(mb_allocation_map_type = = 3)		
for(i = 0; i < num_slice_groups_minus1; i++) {		
top_left_mb	1	u(v)
bottom_right_mb	1	u(v)
}		
else if(mb_allocation_map_type = = 4 || mb_allocation_map_type = = 5 || mb_allocation_map_type = = 6) {		
slice_group_change_direction	1	u(1)
slice_group_change_rate_minus1	1	ue(v)
}		
}		
num_ref_idx_l0_active_minus1	1	ue(v)
num_ref_idx_l1_active_minus1	1	ue(v)
weighted_pred_flag	1	u(1)
weighted_bipred_explicit_flag	1	u(1)
weighted_bipred_implicit_flag	1	u(1)
slice_qp_minus26 /* relative to 26 */	1	se(v)
slice_qp_s_minus26 /* relative to 26 */	1	se(v)
redundant_slice_flag	1	u(1)
vui_pic_parameters_flag	1	u(1)
if(vui_pic_parameters_flag) {		
vui_pic_parameters( )	1	
}		
rbsp_trailing_bits( )	1	
}		

Id., 36 (annotated).

72. The picture parameter syntax in Table 7.3.2.2 includes a value "num_slice_groups_minus1" (annotated with green). *Id.* This parameter indicates the number of slice groups in a picture. JVT-E022d7 discloses that "[i]f num_slice_groups_minusl is zero, all slices of the picture belong to the same slice group [and if] slice_groups_minusl is greater than zero, flexible macroblock ordering is in

use." *Id.*, 54. A POSITA would, therefore, have understood that zero and non-zero numbers are parameter values for the "num_slice_groups_minusl" parameter, which pertains to the picture. The picture parameter syntax in Table 7.3.2.2 also includes a parameter "Num_ref_idx_l0_active_minus1" (annotated with green) which "specifies the number of reference pictures minus l in the reference list 0 that are used to decode the picture." *Id.*, 55. This parameter value of this parameter thereby indicates a maximum number of reference frames that can be used for prediction in the video sequences. The logic for how this parameter is used is described in JVT-E022d7 relating to how macroblocks, which are part of a picture, are predicted. *Id.*, 150 ("if(num_ref_idx_l0_active_minus1 > 0 && mb_partition_pred_mode(mb_type, i) ! = Pred_L1"). A POSITA would have understood that this parameter value pertains to the picture, because it is used in the decoding of a part of a picture, the macroblock, and is listed under "picture parameters." The sequence parameter syntax in Table 7.3.2.2 includes a value "weighted_pred_flag" (annotated with green). The JVT-E022d7 describes the use of this parameter as follows: "weighted_pred_flag equal to zero indicates that weighted prediction is not applied to P and SP slices. weighted_pred_flag equal to one indicates that weighted prediction is applied to P and SP slices" *Id.*, 55. In other words, "weighted_pred_flag" is used in decoding of slices, which are part of a picture. *Id.*, 40. The weighted prediction, which is a technique used to improve coding efficiency in video decoding, is applied in the video coding in accordance with this value. A POSITA would have understood the "weighted_pred_flag" parameter to contain a value that pertains to the picture, because it relates to decoding the slices that are part of the picture and is listed under "picture parameters."

73. In my opinion, a POSITA would have understood that "recognizing, in the decoder" as recited in 6[b] encompasses a decoder parsing the incoming bitstream and identifying a "picture parameter set." In my opinion, JVT-E022d7 performs such "recognizing" because, depending on the implementation, its decoder extracts data elements (such as the picture parameter set) from the bitstream based on the syntax described in the document or the picture parameter set is *a priori* predefined in the decoder and the decoder recognizes a reference to the picture parameter set from the bitstream. *Id.*, 14–15. A POSITA, in my opinion, would have further understood that this parsing and extraction process also results in "forming" (e.g., creating in local memory) the value associated with the data element, such as, for this limitation "at least one first picture parameter value pertaining to a picture using the parameter set."

74. In my opinion, JVT-E022d7 discloses decoding pictures using, for example, a number of slice groups in a picture, a number of reference pictures, and a flag number for weighted prediction ("forming at least one first picture parameter value pertaining to a picture using the parameter set").

d. 6[c]: forming, in the decoder, at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

75. In my opinion, JVT-E022d7 discloses or suggests 6[c]. Preliminarily, JVT-E022d7 teaches "pictures" as well as "slice headers." Specifically, JVT-E022d7

discloses that a "picture" is a "field or a frame" where a "frame contains sampled and quantized luma and chroma data of all rows of a video signal frame. A frame consists of two fields, a top field, and a bottom field." *Id.*, 20–21. A slice group is "[a] sub-set of the macroblocks of a picture." *Id.*, 23. Thus, all slices are part of a picture that is either a frame or a field. A slice header is a "part of a coded slice containing the coded representation of data elements pertaining to the slice data that follow the slice header." *Id*.

76. It is my opinion that JVT-E022d7 also discloses or suggests "forming, in the decoder, at least one second picture parameter value using information of a slice header." JVT-E022d7 provides examples of parameters for a slice header syntax in Table 7.3.3.

frame_num

7.3.3 Slice header syntax

slice_header( ) {	Category	Descriptor
pic_parameter_set_id	4	ue(v)
frame_num	4	u(v)
pic_structure	4	ue(v)
first_mb_in_slice	4	u(v)
slice_type_idc	4	ue(v)
if(pic_order_cnt_type = = 0)		
pic_order_cnt	4	u(v)
else if(pic_order_cnt_type = = 1)		
delta_pic_order_cnt	4	se(v)
if(redundant_slice_flag)		
redundant_pic_cnt	4	ue(v)
if(slice_type_idc = = BiPred)		
direct_spatial_mv_pred_flag	4	u(1)
num_ref_idx_active_override_flag	4	u(1)
if(num_ref_idx_active_override_flag) {		
if(slice_type_idc = = Pred || slice_type_idc = = SPred || slice_type_idc = = BiPred) {		
num_ref_idx_l0_active_minus1	4	ue(v)
if(slice_type_idc = = BiPred)		
num_ref_idx_l1_active_minus1	4	ue(v)
}		
}		

Id., 40 (annotated).

77. Table 7.3.3 includes a parameter "frame_num" (annotated with blue). *Id.*

In accordance with slice header semantics, "frame_num labels the frame" and "frame_num serves as a unique ID for each frame stored in the reference picture buffer." *Id.*, 57. JVT-E022d7 adds that "frame_num shall be incremented by 1 for each coded picture in decoding order." *Id.* This demonstrates that the frame number refers to a value that changes only from picture to picture. Given the structural relationship of picture, frame, and slices, a POSITA would have understood that the value of "frame_num," which serves as a unique identifier for each frame, would be the same for all the slice headers of a frame (i.e., a frame corresponding to a picture). Further, JVT-E022d7 teaches that the frame_num is the element that identifies the frame, and cannot be "equal to any other [frame] among the short term frames in the reference picture buffer." *Id.*, 57. In fact, "[a] decoder which encounters a frame number on a current frame having a value equal to the frame number of some other short-term stored frame in the reference picture buffer should treat this condition as an error." *Id.* In other words, the value of frame_num in JVT-E022d7's cannot change within the frame, because if it did change the decoder would end up decoding the slice into the wrong frame, resulting in an image that did not match what was originally coded. Therefore, JVT-E022d7's disclosure of a slice header that identifies the current frame teaches "one second picture parameter value remaining unchanged at least in all slice headers of one picture," as recited in 6[c].

78. A POSITA, in my opinion, would have understood that "forming, in the decoder, at least one second picture parameter value using information of a slice header"

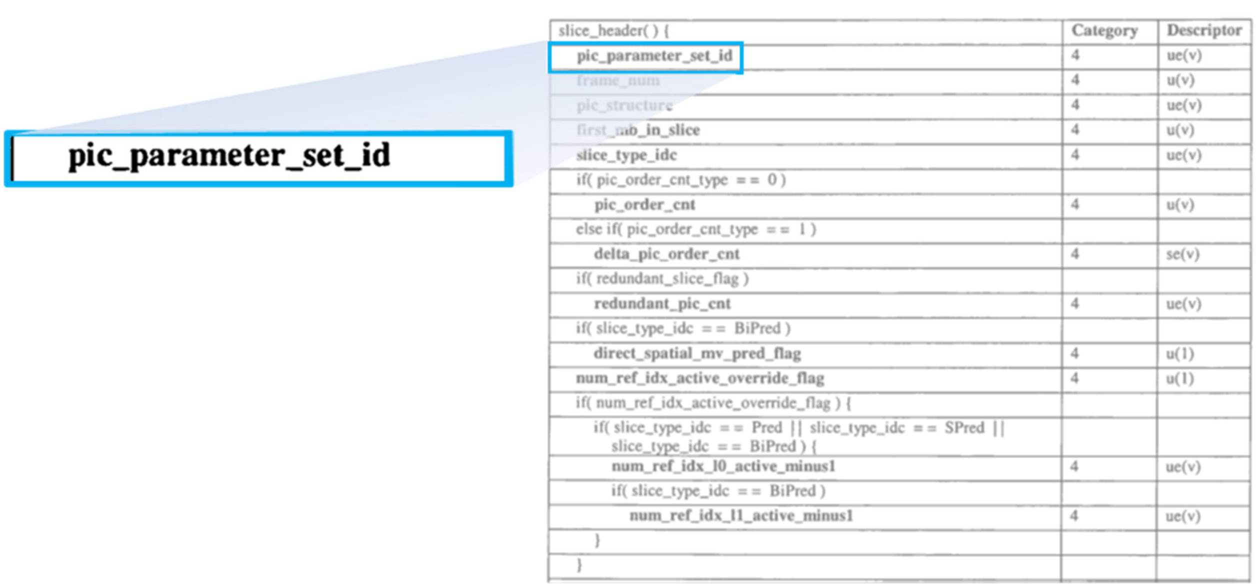
as recited by this limitation encompasses parsing the incoming bitstream and identifying a second picture parameter value in the recited "slice header" ("using information of a slice header") and saving that value in local memory ("forming"). A POSITA would have understood JVT-E022d7 teaches this "forming" limitation for the same reasons I discuss above, including that it describes extracting data elements (such as the slice header) from the bitstream based on the syntax described in the document and because the value of frame_num is saved in local memory along with the reference picture. *Id.*, 14–15.

e. 6[d]: using, in the decoder, the at least one second picture parameter value in decoding.

79. In my opinion, JVT-E022d7 discloses or suggests 6[d]. JVT-E022d7 discloses the decoder uses the value of the "frame_num" parameter (i.e., "the at least one second picture parameter value") in decoding because it states that "frame_num shall be incremented by 1 for each coded picture in decoding order…[t]he decoding order of primary coded pictures shall be non-decreasing in frame number order… frame_num serves as a unique ID for each frame stored in the reference picture buffer [and that a] decoder…encounters a frame number." *Id.*, 57. In my opinion, a POSITA would have understood that frame_num, which is a unique ID for a given frame, when present, is used in decoding a sequence of frames in order to identify each of the framesbecause this unique ID is used to properly index and decode each frame in a given sequence and is therefore used in the decoder for decoding when present.

3. Claim 7: The method according to claim 6, wherein the slice header is included with a reference to a picture parameter set.

80. It is my opinion that JVT-E022d7 discloses or suggests claim 7. Claim 7 depends from claim 6, and Section IX.A.2 explains my opinion that JVT-E022d7 discloses or suggests the "method according to claim 6." As discussed in Section IX.A.2.d for 6[c], JVT-E022d7 discloses that a slice header is "[a] part of a coded slice containing the coded representation of data elements pertaining to the slice data that follow the slice header." *Id.*, 23. JVT-E022d7 also discloses that one of the data elements in the slice header syntax is "pic_parameter_set_id" (annotated with blue below). *Id.*, 40. The parameter "pic_parameter_set_id" is one of the data elements in a picture parameter set syntax, as shown in Table 7.3.2.2. *Id.*, 36. This data element identifies a specific picture parameter set within the bitstream. Therefore, JVT-E022d7's slice header includes "pic_parameter_set_id," which is a reference to a picture parameter set.



7.3.3 Slice header syntax

slice_header( ) {	Category	Descriptor
pic_parameter_set_id	4	ue(v)
frame_num	4	u(v)
pic_structure	4	ue(v)
first_mb_in_slice	4	u(v)
slice_type_idc	4	ue(v)
if(pic_order_cnt_type = = 0)		
pic_order_cnt	4	u(v)
else if(pic_order_cnt_type = = 1)		
delta_pic_order_cnt	4	se(v)
if(redundant_slice_flag)		
redundant_pic_cnt	4	ue(v)
if(slice_type_idc = = BiPred)		
direct_spatial_mv_pred_flag	4	u(1)
num_ref_idx_active_override_flag	4	u(1)
if(num_ref_idx_active_override_flag) {		
if(slice_type_idc = = Pred || slice_type_idc = = SPred || slice_type_idc = = BiPred) {		
num_ref_idx_l0_active_minus1	4	ue(v)
if(slice_type_idc = = BiPred)		
num_ref_idx_l1_active_minus1	4	ue(v)
}		
}		

Id., 40.

4. Claim 8: The method according to claim 6, wherein the picture parameter set is included with a reference to a sequence parameter set.

81. It is my opinion that JVT-E022d7 discloses or suggests claim 8. Claim 8 depends from claim 6, and Section IX.A.2 explains my opinion that JVT-E022d7 discloses or suggests the "method according to claim 6." As discussed in Section IX.A.2.c for 6[b], Table 7.3.2.2 of JVT-E022d7 discloses that one of the data elements in a picture parameter syntax is "seq_parameter_set_id" (annotated with green below). *Id.*, 36. Further, JVT-E022d7 discloses that "seq_parameter_set_id identifies the sequence parameter set to be referred" in a sequence. *Id.*, 52. Therefore, in my opinion, JVT-E022d7's picture parameter set includes "seq_parameter_set_id," which is a reference to a sequence parameter set.

seq_parameter_set_id

7.3.2.2 Picture parameter set RBSP syntax

pic_parameter_set_rbsp( ) {	Category	Descriptor
pic_parameter_set_id	1	ue(v)
seq_parameter_set_id	1	ue(v)
entropy_coding_mode	1	ue(v)
motion_resolution	1	ue(v)
adaptive_block_size_transform_flag	1	u(1)
num_slice_groups_minus1	1	ue(v)
if(num_slice_groups_minus1 > 0) {		
mb_allocation_map_type	1	ue(v)
if(mb_allocation_map_type = = 0)		
for(i = 0; i <= num_slice_groups_minus1; i++)		
run_length	1	ue(v)
else if(mb_allocation_map_type = = 2)		
for(i = 0; i < num_mbs_in_pic; i++)		
slice_group_id [Ed. Note: how many bits ?]	1	u(v)
else if(mb_allocation_map_type = = 3)		
for(i = 0; i < num_slice_groups_minus1; i++) {		
top_left_mb	1	u(v)
bottom_right_mb	1	u(v)
}		
else if(mb_allocation_map_type = = 4 || mb_allocation_map_type = = 5 || mb_allocation_map_type = = 6) {		
slice_group_change_direction	1	u(1)
slice_group_change_rate_minus1	1	ue(v)
}		
}		
num_ref_idx_l0_active_minus1	1	ue(v)
num_ref_idx_l1_active_minus1	1	ue(v)
weighted_pred_flag	1	u(1)
weighted_bipred_explicit_flag	1	u(1)
weighted_bipred_implicit_flag	1	u(1)
slice_qp_minus26 /* relative to 26 */	1	se(v)
slice_qp_s_minus26 /* relative to 26 */	1	se(v)
redundant_slice_flag	1	u(1)
vui_pic_parameters_flag	1	u(1)
if(vui_pic_parameters_flag) {		
vui_pic_parameters( )	1	
}		
rbsp_trailing_bits( )	1	
}		

Id., 36 (annotated).

5. Claim 9: The method according to claim 6, wherein the second picture parameter value is selected from a group of information of a picture order count and a frame number.

82. It is my opinion that JVT-E022d7 discloses or suggests claim 9. Claim 9 depends from claim 6, and Section IX.A.2 explains my opinion that JVT-E022d7 discloses or suggests the "method according to claim 6." As discussed in Section IX.A.2.d for 6[c], it is my opinion that the value of JVT-E022d7's "frame_num" parameter teaches the recites "second picture parameter value." Parameter ("frame_num") identifies the frame number and so is "selected from a group of

information of a picture order count and a frame number," as recited in claim 9's additional limitation. *Id.*, 40, 57.

6. Claim 11

a. 11[pre]: A decoder for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, the decoder comprising:

83. If limiting, in my opinion, JVT-E022d7 discloses or suggests the preamble. As discussed in Section IX.A.2.a for 6[pre], it is my opinion that JVT-E022d7 discloses or suggests "decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices." It is my opinion that JVT-E022d7 further discloses that such decoding is done by a decoder. *Id.*, 17 ("How to read this specification…clause 6 (Source, coded, decoded, and output data formats) should be read for the relationship of the sou[r]ce, input, and output of the decoder"), 145 ("Figure 11-1 – A block diagram of a conceptual decoder for non-intra coded macroblocks in SP slices"), Fig. 11-1.

b. 11[a]: a processor configured for:

84. In my opinion, JVT-E022d7 discloses or suggests 11[a]. A POSITA would have understood that JVT-E022d7 discloses a processor for decoding a video. JVT-E022d7 describes that "[t]he use of this Recommendation | In[]ternational Standard allows motion video to be manipulated as a form of computer data and to be stored on various storage media." *Id.*, 14. A processor is a functional unit in a computer that interprets and executes instructions, including for data processing. EX-1019, 5

("Central processing unit (CPU): Controls the operation of the computer and performs its data processing functions; often simply referred to as **processor**."), 8 (defining processor as "In a computer, a functional unit that interprets and executes instructions. A processor consists of at least an instruction control unit and an arithmetic unit."). In my opinion, a POSITA would have understood, and found it obvious, that a processor performs the manipulation (including decoding) of motion video in the form of computer data so that the manipulated computer data can be stored on various storage media, as is described by JVT-E022d7. JVT-E022d7, 14; *see also* EX-1015, 15 (describing processors of full motion video). In the context of JVT-E022d7, a POSITA, in my opinion, would have understood a processor performs the steps in 11[b]-11[e] as discussed below.

c. 11[b]: recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set for a picture;

85. As discussed in Section IX.A.2.b for 6[a], it is my opinion that JVT-E022d7 discloses or suggests "recognizing, in a decoder, a sequence parameter set and forming at least one sequence parameter pertaining to a sequence using the parameter set." 11[b] is substantially similar to 6[a] except 11[b] recites "recognizing a sequence parameter set for a sequence of pictures [and] using the parameter set for a picture." It is my opinion that these additions in 11[b] are also taught by JVT-E022d7 because JVT-E022d7 discloses that a "video source that is represented by the bitstream is a sequence of frames and/<u>or</u> fields (called collectively pictures) in decoding order." JVT-E022d7,

27. In my opinion, a POSITA would have understood that the sequence parameter is "for a sequence of pictures," as recited by this limitation, because a video in a bitstream is a sequence of pictures in a decoding order and the sequence parameters apply to one or more pictures in a coded video sequence. And in my opinion, a POSITA would have also understood that at least some of the parameters in the sequence parameter set disclosed by JVT-E022d7, including the picture order count type ("pic_order_cnt_type," as explained in 6[a]) would be "for a picture," as required by this limitation because that parameter is used for displaying pictures in a picture sequence in correct order and is therefore defined for each picture.

d. 11[c]: recognizing a picture parameter set and forming at least one first picture parameter value pertaining to a picture using the parameter set;

86. In my opinion, JVT-E022d7 discloses or suggests 11[c] for substantially the same reasons as 6[b]. *See* Section IX.A.2.c.

e. 11[d]: forming at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

87. In my opinion, JVT-E022d7 discloses or suggests 11[d] for substantially the same reasons as 6[c]. *See* Section IX.A.2.d.

f. 11[e]: using the at least one second picture parameter value in decoding.

88. In my opinion, JVT-E022d7 discloses or suggests 11[e] for substantially the same reasons as 6[d]. *See* Section IX.A.2.e.

7. Claim 15

a. 15[pre]: A receiving device comprising a decoder for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, the decoder comprising:

89. If limiting, in my opinion, JVT-E022d7 discloses or suggests 15[pre]. As discussed in Section IX.A.6.a for 11[pre], it is my opinion that JVT-E022d7 discloses or suggests: "A decoder for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices." 15[pre] adds only a "receiving device" comprising such a decoder. At the relevant time, it was conventional for a receiving device to include the decoder, which would have been understood by a POSITA. To this effect, a decoder is utilized at the receiver end to reconstruct the pixel intensities from the received bitstream created by the encoder. For example, Tekalp envisions using compressed digital video formats for a number of devices, including "all-digital TV," "videophone," and "multimediastations." EX-1015, 8. Tekalp further shows that a decoder was well known as reversing the operations of the encoder. *Id.*, 12. As another example of a what was known in the art, Au, discussed below, shows that devices including decoders were well known in the art at the time of the purported invention. Au, 6:66–7:3 ("exchange video data **12** between monitors **13**, formatted as a compressed bit stream **15**"). This is illustrated in Figure 1 of Au, reproduced below.

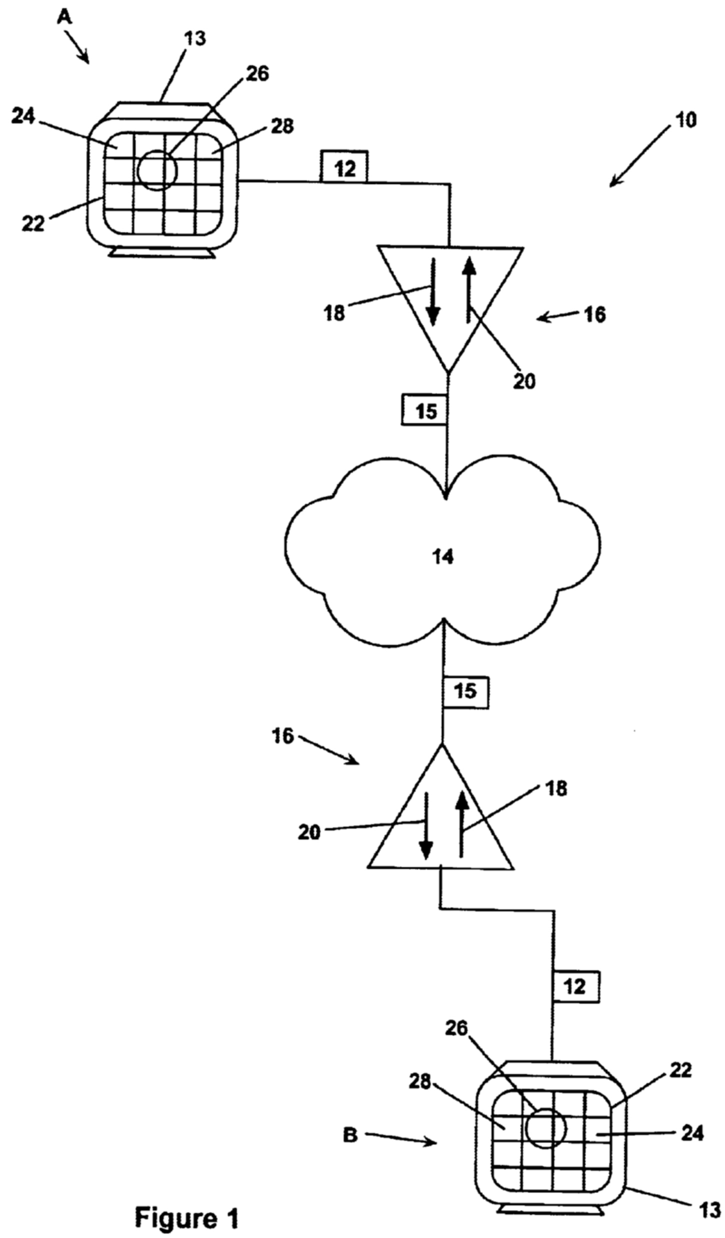


Figure 1

90. Accordingly, JVT-E022d7 discloses or suggests this additional requirement. Specifically, JVT-E022d7 "allows motion video to be manipulated as a form of computer data and to be stored on various storage media, transmitted and received over existing and future networks." JVT-E022d7, 14. In my opinion, a POSITA would have understood this discloses, or at least suggests, a "receiving device"

because it receives and stores data, and a POSITA would have also understood decoding in JVT-E022d7 is performed in such a device because the received data is what is decoded.

b. 15[a]: a processor configured for:

91. In my opinion, JVT-E022d7 discloses or suggests 15[a] for substantially the same reasons as 11[a]. *See* Section IX.A.6.b.

c. 15[b]: recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set;

92. In my opinion, JVT-E022d7 discloses or suggests 15[b] for substantially the same reasons as 11[b]. *See* Section IX.A.6.c.

d. 15[c]: recognizing a picture parameter set for a picture and forming at least one first picture parameter value pertaining to a picture using the parameter set;

93. In my opinion, JVT-E022d7 discloses or suggests 15[c] for substantially the same reasons as 11[c]. *See* Section IX.A.6.d.

e. 15[d]: forming at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

94. In my opinion, JVT-E022d7 discloses or suggests 15[d] for substantially the same reasons as 11[d]. *See* Section IX.A.6.e.

f. 15[e]: using the at least one second picture parameter value in decoding.

95. In my opinion, JVT-E022d7 discloses or suggests 15[e] substantially the same reasons as 11[e]. *See* Section IX.A.6.f.

8. Claim 21

a. 21[pre]: A computer readable storage medium embedded with a computer program comprising programming code for execution on a processor, the programming code for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, and the programming code for:

96. If limiting, in my opinion, JVT-E022d7 discloses or suggests the preamble. As discussed in Section IX.A.2.a for 6[pre], it is my opinion that JVT-E022d7 discloses or suggests "decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices." Section IX.A.6.b for 11[a] explains my opinion that JVT-E022d7 discloses or suggests a processor configured to perform the recited steps that follow. 21[pre] adds only the requirement of a "computer readable storage medium embedded with a computer program comprising programming code for execution on a processor" and that program is for the "decoding" similarly recited in 6[pre]. In my opinion, JVT-E022d7 discloses this addition.

97. As discussed in Section IX.A.6.a for 11[pre], JVT-E022d7's decoding is done by a decoder. JVT-E022d7, 17 ("How to read this specification…clause 6 (Source, coded, decoded, and output data formats) should be read for the relationship of the sou[r]ce, input, and output of the decoder;"), 145 ("Figure 11-1 – A block diagram of a conceptual decoder for non-intra coded macroblocks in SP slices";), Fig. 11-1. And as discussed in Section IX.A.6.b for 11[a], a POSITA would have understood that JVT-E022d7 discloses a processor for decoding a video. *Id.*, 14 ("The use of this

Recommendation | In[]ternational Standard allows motion video to be manipulated as a form of computer data and to be stored on various storage media"). As described in William Stallings, Computer Organization and Architecture, Designing for Performance, 8th Edition (2010) ("Stallings"), a POSITA would have understood that a computer program is written in a source language (either assembly language, which is assembled into executable code by an assembler, or a higher level language such as C++, which is compiled into executable code). EX-1019, 6. Without such a program, the processor will not execute any instructions. As further described in Stallings, a program does not reside in main memory all of the time, but is loaded into memory for execution. *Id.*, 7 ("The first step in the creation of an active process is to load a program into main memory…."), 6 (defining "Loader" as "[a] program routine that copies an executable program into memory for execution."). Stallings further describes swapping processes out to disk. *Id.*, 7. Therefore, in my opinion, a POSITA would have understood that such a decoder using a processor would have been implemented with programming code in a computer program stored on a computer readable storage medium.

b. 21[a]: recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set;

98. In my opinion, JVT-E022d7 discloses or suggests 21[a] for substantially the same reasons as 11[b]. *See* Section IX.A.6.c.

c. 21[b]: recognizing a picture parameter set for a picture and forming at least one first picture parameter value pertaining to a picture using the parameter set;

99. In my opinion, JVT-E022d7 discloses or suggests 21[b] for substantially the same reasons as 6[b]. *See* Section IX.A.2.c.

d. 21[c]: forming at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

100. In my opinion, JVT-E022d7 discloses or suggests 21[c] for substantially the same reasons as 6[c]. *See* Section IX.A.2.d.

e. 21[d]: using the at least one second picture parameter value in decoding.

101. In my opinion, JVT-E022d7 discloses or suggests 21[d] for substantially the same reasons as 6[d]. *See* Section IX.A.2.e.

9. Claim 23

a. 23[pre]: A decoder for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, the decoder comprising:

102. If limiting, in my opinion, JVT-E022d7 discloses or suggests 23[pre] for substantially the same reasons as 11[pre]. *See* IX.A.6.a.

b. 23[a]: means for recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set for a picture;

103. In my opinion, JVT-E022d7 discloses or suggests 23[a] regardless of whether it is construed as means-plus-function (MPF).

104. In my opinion, JVT-E022d7 teaches the function as discussed in Section

IX.A.6.c for 11[b].

105. If this term is construed as MPF, in my opinion, JVT-E022d7 discloses or suggests the structure discussed in Section VIII.A ("a decoder as described in the '818 patent specification in figure 4 and equivalents thereof"). As shown below, Figure 4 of the '818 patent discloses a structure that includes a processor, decoding buffer, program code (the decoding buffer and program code both stored in memory).

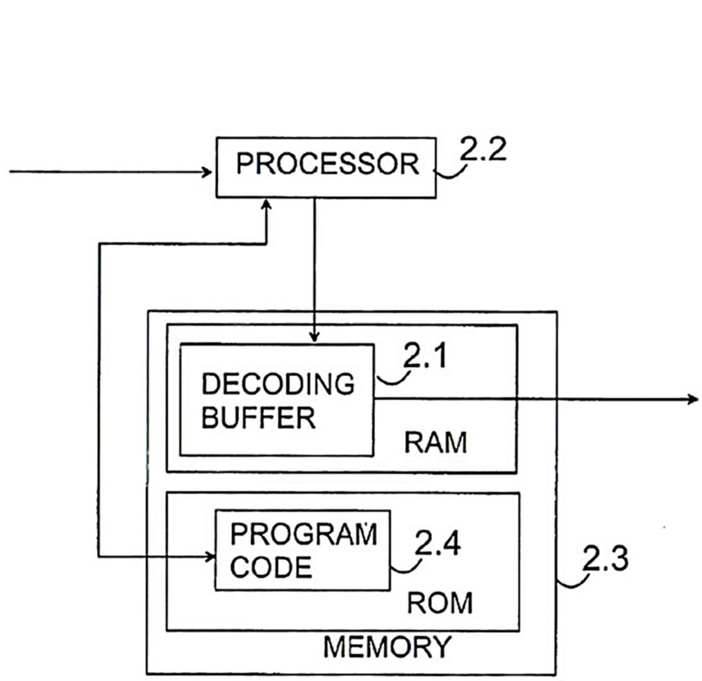


Fig. 4

106. As discussed in Section IX.A.6.c for 11[b], it is my opinion that JVT-E022d7 discloses that its decoding is performed by a decoder. JVT-E022d7, 17 ("How

to read this specification…clause 6 (Source, coded, decoded, and output data formats) should be read for the relationship of the sou[r]ce, input, and output of the decoder;"), 145 ("Figure 11-1 – A block diagram of a conceptual decoder for non-intra coded macroblocks in SP slices"), Fig. 11-1. Section IX.A.6.b for 11[a] explains my opinion that JVT-E022d7 discloses or suggests a processor configured to perform the recited steps that follow. Section IX.A.8.a for 21[pre] explains that a POSITA, in my opinion, would have understood, or found it obvious, that JVT-E022d7's decoder uses a processor implemented with programming code in a computer program stored on a computer readable storage medium. JVT-E022d7's decoder also includes a number of buffers, including "short and long term buffers" and a "reference picture buffer." *Id.*, 79–80. JVT-E022d7 discloses these buffers are managed using "Adaptive Memory Control." For these reasons, a POSITA, in my opinion, would have understood JVT-E022d7 discloses a decoder with structure the same as, or equivalent to, that of Figure 4 of the '818 patent, and that decoder performs the recited function in 23[a].

c. 23[b]: means for recognizing a picture parameter set and forming at least one first picture parameter value pertaining to a picture using the parameter set;

107. In my opinion, JVT-E022d7 discloses or suggests 23[b] regardless of whether it is construed as MPF.

108. In my opinion, JVT-E022d7 teaches the function as discussed in Section IX.A.2.c for 6[b].

109. If this term is construed as MPF, in my opinion, JVT-E022d7's decoder

discloses or suggests the structure discussed in Section VIII.A(“a decoder as described in the ’818 patent specification in figure 4 and equivalents thereof”) for the same reasons discussed in Section IX.A.9.b for 23[a] above and that decoder performs the recited function in 23[b].

d. 23[c]: means for forming at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

110. In my opinion, JVT-E022d7 discloses or suggests 23[c] regardless of whether it is construed as MPF.

111. In my opinion, JVT-E022d7 teaches the function as discussed in Section IX.A.2.d for 6[c].

112. If this term is construed as MPF, in my opinion, JVT-E022d7’s decoder discloses or suggests the structure discussed in Section VIII.A(“a decoder as described in the ’818 patent specification in figure 4 and equivalents thereof”) for the same reasons discussed in Section IX.A.9.b for 23[a] above and that decoder performs the recited function in 23[c].

e. 23[d]: means for using the at least one second picture parameter value in decoding.

113. In my opinion, JVT-E022d7 discloses or suggests 23[d] regardless of whether it is construed as MPF.

114. In my opinion, JVT-E022d7 teaches the function as discussed in Section IX.A.2.e for 6[d].

115. If this term is construed as MPF, in my opinion, JVT-E022d7’s decoder

discloses or suggests the structure discussed in Section VIII.A("a decoder as described in the '818 patent specification in Figure 4 and equivalents thereof") for the same reasons discussed in Section IX.A.9.b for 23[a] above and that the decoder performs the recited function in 23[d].

B. Ground 2: Obviousness over Linzer (Claims 6–9, 11, 15, 21, and 23)

1. Linzer

116. Linzer relates to a "method for decoding a digital video bit-stream." Linzer, Abstract. In its "Background of the Invention," Linzer "incorporated by reference" the "entirety" of "document JVT-E022d7 titled 'Editor's Proposed Draft Text Modifications for Joint Video Specification (IUT-T Rec. H.264 ISO/IEC 14496-10 AVC), Draft 7' published Sep. 19, 2002 by the Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG, Berlin, German" (JVT-E022d7, EX-1007, included in LinzerFH at 47–264). Linzer, 1:27–35. It is my understanding that Linzer discloses non-patent publication JVT-E022d7 as background information about the state of the art as if it were fully contained within Linzer. The Linzer applicant filed JVT-E022d7 in its entirety with the Linzer application. EX-1006, 47.

2. Claims 6–9, 11, 15, 21, and 23

117. In addition to being a printed publication under §102(a), it is my understanding that JVT-E022d7's disclosure is also applicable as prior art under §102(e)(2) because it was incorporated by reference in its entirety in Linzer. Therefore, it is my opinion that claims 6–9, 11, 15, 21, and 23 are unpatentable as obvious over

Linzer based on the same disclosures of JVT-E022d7 (incorporated by reference into Linzer) as discussed in Section IX.A.

C. Ground 3: Obviousness over Au (Claims 6–9, 11, 15, 21, and 23)

118. It is my opinion that claims 6–9, 11, 15, 21, and 23 of the ’818 patent are unpatentable as obvious in view of Au, either on its own or in combination with the knowledge of a POSITA.

1. Au

119. Au relates to “a system and method for context adaptive entropy decoding of transform coefficients in compressed video images.” Au, 1:7–9. Part of Au’s system and method involves “a hierarchy of syntax levels, with the 3 main levels being a sequence level 17, a picture (or frame) level 19, and slice level 21.” *Id.*, 8:7–9. Au also discloses “parameter sets,” that “allow[] efficient transmission of infrequently changing data at the sequence 17 and picture level 19 in the H.264 standard.” *Id.*, 8:9–11. Each sequence parameter set “includes values of parameters that will remain unchanged for an entire video sequence, or from one instantaneous decoder refresh (IDR) picture to the next.” *Id.*, 8:13–15. Each picture parameter set “includes values of parameters that will remain unchanged within a coded representation of a picture (frame or field) 22.” *Id.*, 8:20–22. In addition, Au discloses a “slice header,” which contains “a coded representation of data elements 35 that pertain to the decoding of the slice data that follow the slice header 27.” *Id.*, 8:40–42. The hierarchy, parameter sets, and slice header are illustrated in Figure 2 of Au, reproduced below.

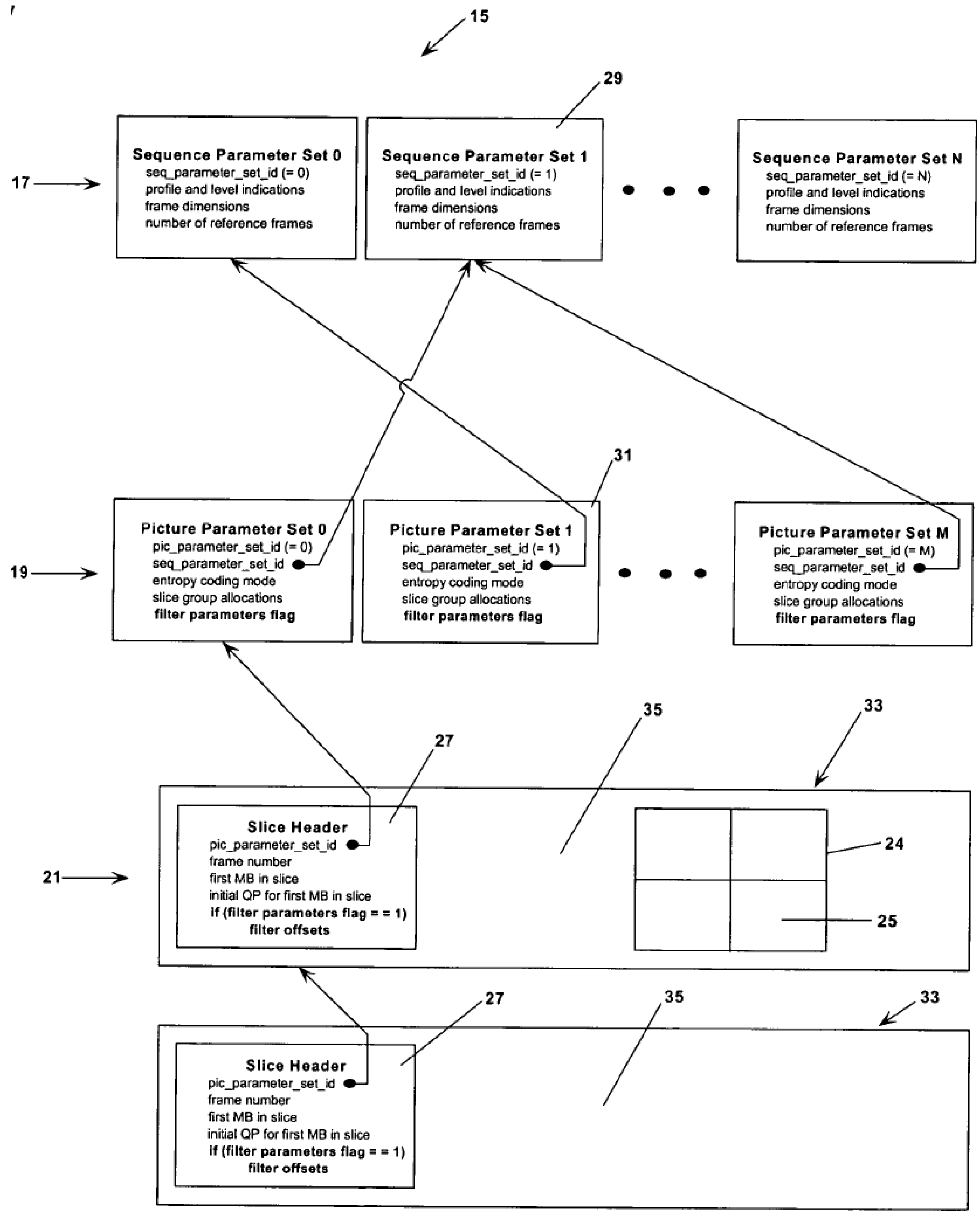


Figure 2

120. As shown in Figure 2, each slice and picture parameter set indicates the

parameter set next up in the hierarchy by including the identifier for that parameter set. *See also id.*, 9:12–14.

121. Au's hierarchy of syntax levels corresponds to the background disclosure in '818 patent, as explained in Section VII.A above (i.e., bitstream includes sequences of pictures; a picture sequence corresponds to group of pictures; a picture is a frame of a field; a picture consists of one or more slices). In my opinion, Au also describes the purported invention of the '818 patent: splitting the parameter sets into the "sequence parameter set," and the "picture parameter set," as shown in Au's Figure 2 hierarchy above, as well as additional picture parameters included in a "slice header" (e.g., Au's "frame_num" included in each slice header, the value of which would not change for all slice headers in a picture, as discussed further below).

2. Claim 6

a. 6[pre]: A method for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, the method comprising:

122. If limiting, in my opinion, Au discloses or suggests 6[pre]. As to "decoding sequences of pictures from a bitstream," Au discloses "[a] method…for block-based motion-compensated decoding." Au, Abstract. Au further discloses a bitstream that is processed by a decoder to "produce reconstructed video images" which correspond to "reconstructed frames." Au, 10:50–53. A POSITA would have understood such reconstruction to be decoding a sequence of pictures from the bitstream because frames are basically synonymous with pictures in the modern digital video

reconstruction. *See* Watkinson, 9 (a "picture" "could be a field or a frame in interlaced systems but only a frame in non-interlaced systems").

123. Au further discloses "parameter sets" that include "values of parameters…." Au, 8:9–19. A POSITA would have understood this to disclose "parameters are defined in a parameter set."

124. As to "each picture comprises information of one or more slices," Au discloses a slice that "contains a picture data **35** representing a sub-set of the macroblocks **24** of the complete picture **22**." *Id.*, 8:36–38. Therefore, each picture comprises information ("data…representing a sub-set of the macroblocks") of one or more slices.

125. In sum, in my opinion, Au discloses processing a bitstream to produce reconstructed video images corresponding to frames ("decoding sequences of pictures from a bitstream"); that bitstream includes sequence parameter sets and picture parameter sets ("parameters…defined in a parameter set") and slices that make up each picture ("each picture comprises information of one or more slices").

b. 6[a]: recognizing, in a decoder, a sequence parameter set and forming at least one sequence parameter pertaining to a sequence using the parameter set;

126. In my opinion, Au discloses or suggests 6[a]. Au discloses that since the video data is transmitted or stored in the form of compressed bitstream, a decoder is needed to decode the bitstream to reconstruct the video. *Id.*, 1:55–57. Au therefore describes a decoder. Au discloses a "sequence parameter set," that includes "values of

parameters that will remain unchanged for an entire video sequence, or from one instantaneous decoder refresh (IDR) picture to the next." *Id.*, 8:11–15. Au discloses examples of parameters in a sequence parameter set, including "frame dimensions and the maximum number of reference frames," as well as "profile and level indications[.]" *Id.*, 8:16–19, Fig. 2. Au discloses that the "active sequence parameters…are used when decoding coded pictures…." *Id.*, 8:30–34, 9:11–14 ("The parameter sets **31** also contain a reference to the sequence parameter sets **29**, which are active for decoding of the pictures **22**.").

127. In my opinion, a POSITA would have understood that "recognizing, in a decoder" as recited by this limitation encompasses a decoder parsing the incoming bitstream and identifying the recited "sequence parameter set." A POSITA would have understood that "forming" as recited by this limitation encompasses the decoder creating in local memory the data element, such as, for this limitation "at least one sequence parameter pertaining to a sequence using the parameter set." A POSITA, in my opinion, would have understood that Au's decoder involves parsing the incoming bitstream to extract ("recognizing") and store ("forming") data elements, including the recited "sequence parameter set" or the sequence parameter set is *a priori* predefined in the decoder and the decoder recognizes a reference to the sequence parameter set from the bitstream described above.

128. Therefore, in my opinion, Au discloses decoding pictures ("in a decoder") using, for example, frame dimensions, a maximum number of reference frames, and

profile and level indications ("forming at least one sequence parameter pertaining to a sequence using the parameter set").

c. 6[b]: recognizing, in the decoder, a picture parameter set and forming at least one first picture parameter value pertaining to a picture using the parameter set;

129. In my opinion, Au discloses or suggests 6[b]. First, Au discloses a "picture parameter set," that includes "values of parameters that will remain unchanged within a coded representation of a picture (frame or field) **22**." *Id.*, 8:20–22. Au discloses examples of parameters in a picture parameter set, including "the entropy coding mode and a flag that specifies whether deblocking filter parameters will be transmitted in the slice headers **27** of the picture **22**[,]" as well as "slice group allocations[.]" *Id.*, 8:23–26, Fig. 2. Au also discloses that "each picture parameter set **31** contains parameter values that pertain to the decoding of the pictures **22** for which the particular parameter set **31** is active (i.e. selected in the slice headers **27** of the picture **22**)." *Id.*, 9:8–12. Au decodes pictures ("in a decoder") using, for example, entropy coding mode, a flag that specifies whether deblocking filter parameters will be transmitted in the slice headers, and slice group allocations ("forming at least one first picture parameter value pertaining to a picture using the parameter set").

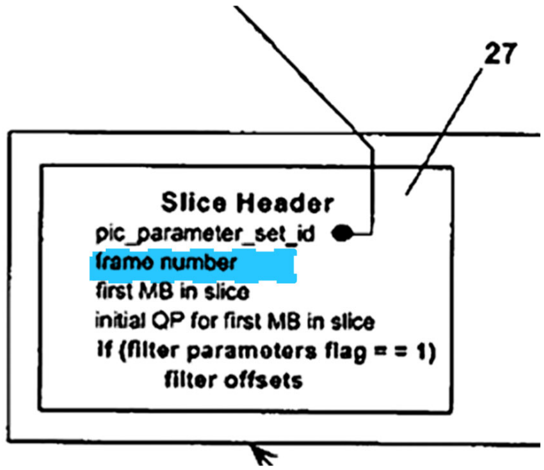
130. In my opinion, a POSITA would have understood that "recognizing, in the decoder" as recited by this limitation encompasses a decoder parsing the incoming bitstream and identifying the recited "picture parameter set." A POSITA would have understood that "forming" as recited by this limitation encompasses the decoder

creating in local memory the value associated with the data element, such as, for this limitation "at least one first picture parameter value pertaining to a picture using the parameter set." In my opinion, a POSITA would have understood that Au's decoder involves parsing the incoming bitstream to extract ("recognizing") and store ("forming") data elements, including the recited "picture parameter set" or the picture parameter set is *a priori* predefined in the decoder and the decoder recognizes a reference to the picture parameter set from the bitstream described above.

131. In my opinion, Au discloses decoding pictures ("in the decoder") using, for example, entropy coding mode, a flag that specifies whether deblocking filter parameters will be transmitted in the slice headers of the picture, and slice group allocations ("forming at least one first picture parameter value pertaining to a picture using the parameter set").

d. 6[c]: forming, in the decoder, at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

132. In my opinion, Au discloses or suggests 6[c]. As an initial matter, Au teaches "pictures" as well as "slice headers." Specifically, Au discloses that a "picture" is a "frame or field." Au, 8:22. Au discloses information of a slice header in the form of "a coded representation of data elements **35** that pertain to the decoding of the slice data that follow the slice header **27**." *Id.*, 8:40–42. Those data elements include "frame number," (annotated with blue) as shown in Figure 2, relevant portion reproduced below.



133. A POSITA would have understood that a frame number is a picture parameter value, and therefore meets the limitation of "at least one second picture parameter value" that is formed by the decoder using information from a slice header.

134. In my opinion, Au also discloses or suggests "forming, in the decoder, at least one second picture parameter value using information of a slice header." A POSITA, in my opinion, would have understood that "forming, in the decoder" as recited by this limitation encompasses a decoder creating in local memory the value associated with the data element in order for it to be utilized in the decoding process, such as, for this limitation "at least one second picture parameter value using information of a slice header." A POSITA would have understood that Au's decoder involves parsing the incoming bitstream to extract and store ("forming") values associated with data elements, including the recited "second picture parameter value" described above.

135. Au also discloses or suggests "the at least one second picture parameter

value remaining unchanged at least in all slice headers of one picture." Au discloses that a slice "contains a picture data **35** representing a sub-set of the macroblocks **24** of the complete picture **22**." Au, 8:36–38. Au discloses that a picture is a "frame or field." *Id.*, 8:22; *see also id.*, 7:25–26. Therefore, in my opinion, Au discloses a picture that is a frame, the data of which is included in slices. In such a case in which frame_num is used, each slice of the picture would have the same frame number in order to properly reconstruct the frame. That frame number, identified above as the "second picture parameter value" limitation, would thus "remain[] unchanged at least in all slices of one picture," as required by this limitation. frame_num would function this way to ensure that all of the slices that belong to a given frame are indexed accordingly in order to spatially position them correctly within the frame for proper reconstruction. Hence, in my opinion, Au's slice header includes frame number ("at least one second picture parameter value using information of a slice header"), which is the same for all slices in the frame ("remaining unchanged at least in all slice headers of one picture").

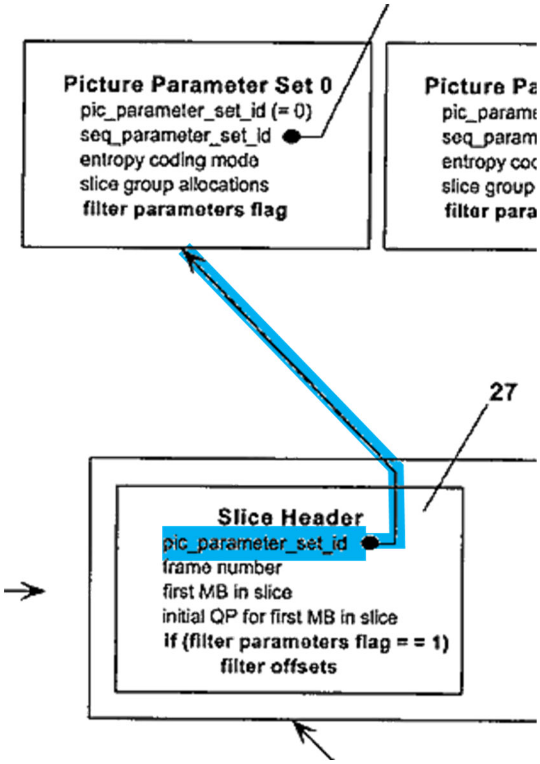
e. 6[d]: using, in the decoder, the at least one second picture parameter value in decoding.

136. In my opinion, Au discloses or suggests 6[d]. Au discloses the information in the slice headers "pertain[s] to the decoding of the slice data that follow the slice header **27**." *Id.*, 8:40–43. In my opinion, Au discloses using a frame number in decoding video sequences because the frame number value is a unique identifier for each picture. A decoder would use the frame in the slice header to know which frame the slice belongs to and is currently being decoded in the picture buffer. In my opinion,

a POSITA would have understood this to disclose using each piece of information in the slice header, including the frame number, identified above as the "second picture parameter value" limitation, in decoding.

3. Claim 7: The method according to claim 6, wherein the slice header is included with a reference to a picture parameter set.

137. It is my opinion that Au discloses or suggests claim 7. Claim 7 depends from claim 6, and Section IX.C.2 explains that Au, in my opinion, discloses or suggests the "method according to claim 6." As discussed in Section IX.C.2.d for 6[c], Au discloses that one of the data elements in the slice header "contains a reference to a valid picture parameter set **31**, which specifies the picture parameter values…to be used when decoding..." Au, 8:43–48. This is illustrated in Figure 2 (relevant section reproduced below, with annotations), which shows the slice header includes an element "pic_parameter_set_id" with an arrow pointing to a picture parameter set.

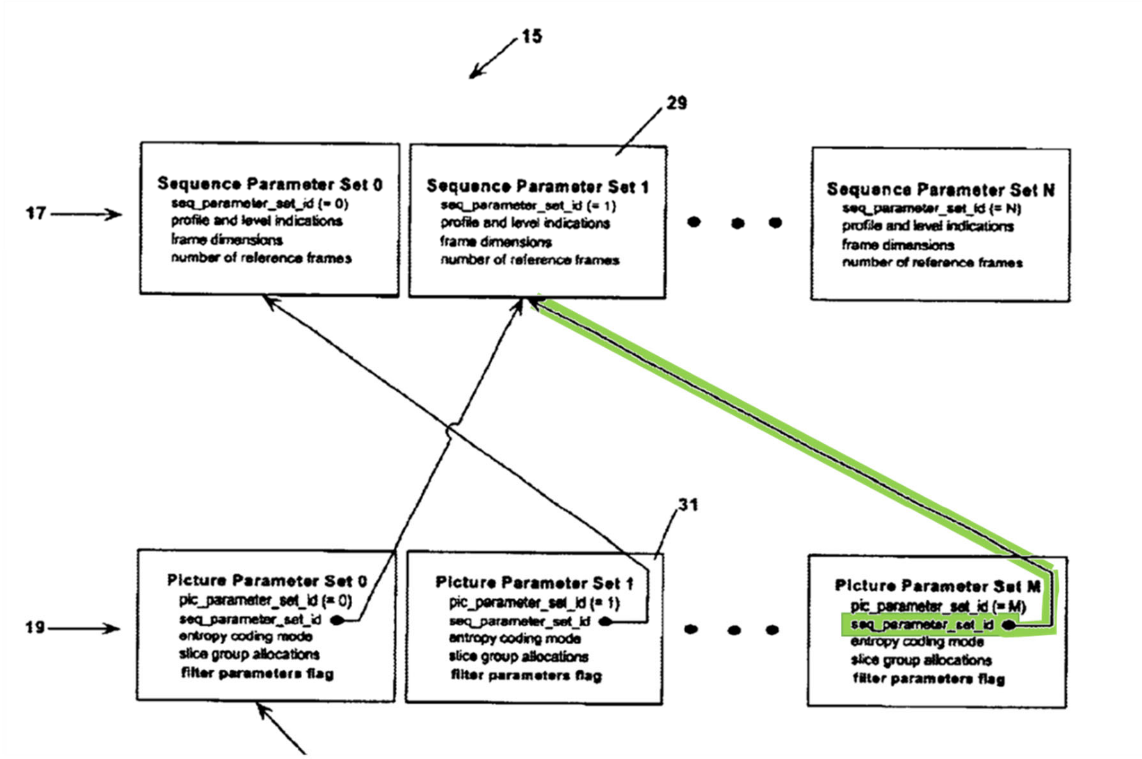


138. Therefore, Au's slice header includes "pic_parameter_set_id," which is a reference to a picture parameter set.

4. Claim 8: The method according to claim 6, wherein the picture parameter set is included with a reference to a sequence parameter set.

139. It is my opinion that Au discloses or suggests claim 8. Claim 8 depends from claim 6, and Section IX.C.2 explains that Au, in my opinion, discloses or suggests the "method according to claim 6." As discussed in Section IX.C.2.c for 6[b], Au discloses that "[e]ach picture parameter set **31**…refers to the unique ID of a valid sequence parameter set **29**, which selects the active sequence parameters that are used when decoding coded pictures **22** that use the particular picture parameter set [**31**]" Au, 8:30–34. This is illustrated in Figure 2 (relevant section reproduced below, with annotations), which shows each picture parameter set includes an element

"seq_parameter_set_id" with an arrow pointing to a sequence parameter set.



140. Therefore, Au's picture parameter set includes "seq_parameter_set_id," which is a reference to a sequence parameter set.

5. Claim 9: The method according to claim 6, wherein the second picture parameter value is selected from a group of information of a picture order count and a frame number.

141. It is my opinion that Au discloses or suggests claim 9. Claim 9 depends from claim 6, and Section IX.C.2 explains that Au, in my opinion, discloses or suggests the "method according to claim 6." As discussed in Section IX.C.2.d for 6[c], it is my opinion that the frame number of Au teaches the recited "second picture parameter value." The "frame number" is a parameter that identifies the appropriate frame number in a given sequence and so is "selected from a group of information of a picture order count and a frame number," as recited in claim 9's additional limitation.

6. Claim 11

a. 11[pre]: A decoder for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, the decoder comprising:

142. If limiting, in my opinion, Au discloses or suggests the preamble. As discussed in Section IX.C.2.a for 6[pre], Au, in my opinion, discloses or suggests "decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices." It is my opinion that Au further discloses that such decoding is done by a decoder. Au, 6:54–55 ("FIG. 3 shows further details of a decoder of the system of FIG. 1;"), 10:41–46 ("The decoder **20** (see FIG. **3**) of the receiving participant A,B processes the received bit stream **15** and then reconstructs the coded frame…."), Fig. 3.

b. 11[a]: a processor configured for:

143. In my opinion, Au discloses or suggests 11[a]. Au discloses "the processors **16** coordinate the display of successive frames **22** on the monitors **13**, as the video data **12** is communicated between the participants A, B during a video conference." *Id.*, 7:11–14. This is illustrated in Figure 1, reproduced below.

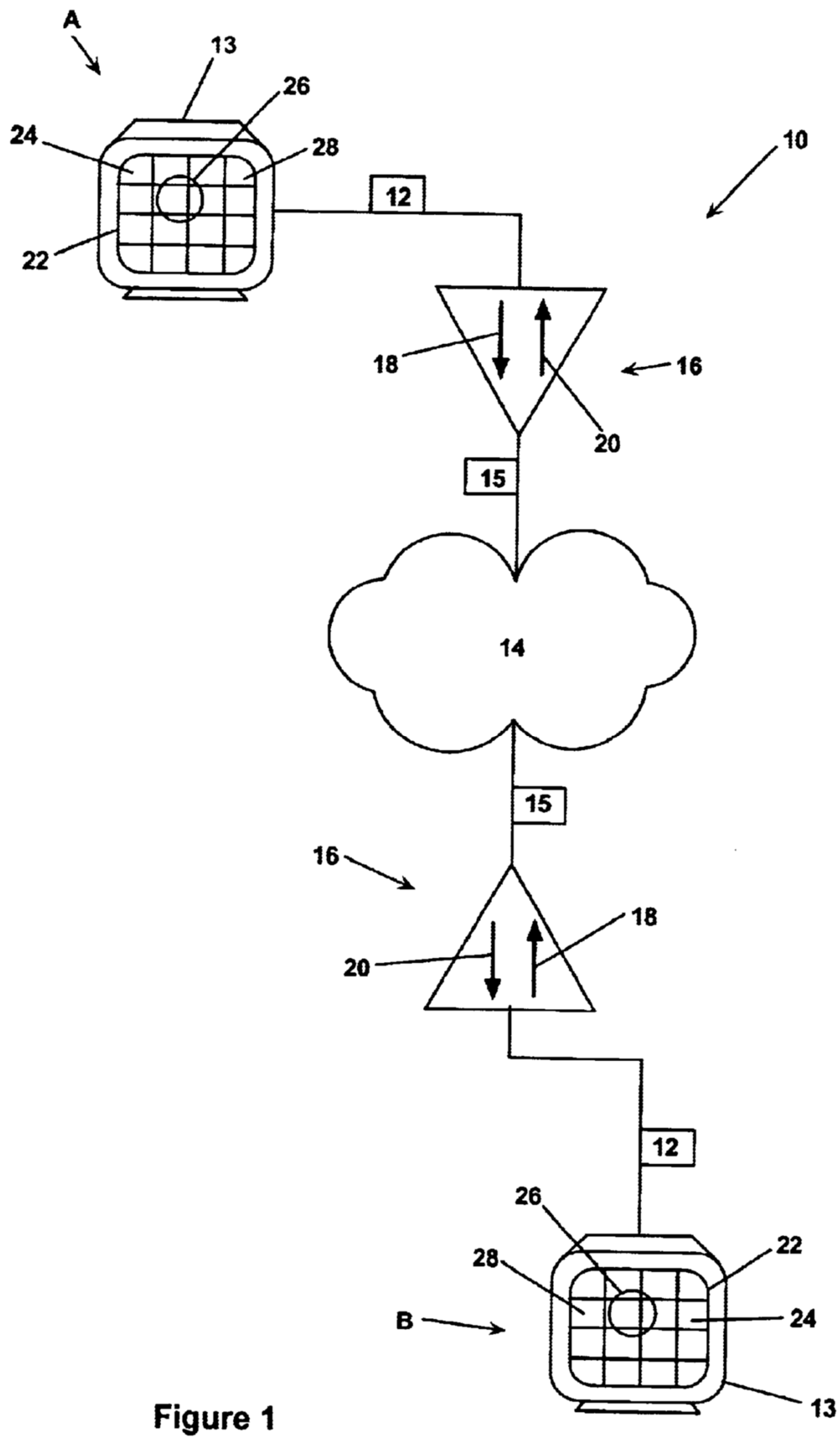


Figure 1

Au's decoder 20 comprises a processor 16, as required by this limitation. If this limitation requires a processor to be dedicated to decoding, it is my opinion that it would have been obvious to a POSITA to implement a decoder in a dedicated processor. This would have involved routine engineering and would have ensured the decoder would be reliable and have ample processing power, as it would not have had to share processing power for other functionality. A dedicated processor would also provide performance increases, as it could be implemented using hardware specifically designed for the task of decoding, including parallel processing.

c. 11[b]: recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set for a picture;

144. As discussed in Section VII.C.2.b 6[a], in my opinion, Au discloses "recognizing, in a decoder, a sequence parameter set and forming at least one sequence parameter pertaining to a sequence using the parameter set." 11[b] is substantially similar to 6[a] except that 11[b] recites "recognizing a sequence parameter set for a sequence of pictures [and] using the parameter set for a picture." It is my opinion that these additions in 11[b] are also taught by Au because Au discloses that a sequence parameter set "includes values of parameters that will remain unchanged for an entire video sequence, or from one instantaneous decoder refresh (IDR) picture to the next." Au, 8:12–15. A POSITA would have understood that both "an entire video sequence" and a sequence "from one [IDR] picture to the next" would each be a "sequence of pictures" and thus the sequence parameter set is "for a sequence of pictures," as recited by this limitation. A POSITA would have also understood that at least some of the parameters in the sequence parameter set disclosed by Au, including frame dimensions would be "for the picture," as required by this limitation because that parameter is used

for displaying pictures in the correct dimensions. *Id.*, 8:16–19, Fig. 2.

d. 11[c]: recognizing a picture parameter set and forming at least one first picture parameter value pertaining to a picture using the parameter set;

145. In my opinion, Au discloses or suggests 11[c] for substantially the same reasons as 6[b]. *See* Section IX.C.2.c.

e. 11[d]: forming at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

146. In my opinion, Au discloses or suggests 11[d] for substantially the same reasons as 6[c]. *See* Section IX.C.2.d.

f. 11[e]: using the at least one second picture parameter value in decoding.

147. In my opinion, Au discloses or suggests 11[e] for substantially the same reasons as 6[d]. *See* Section IX.C.2.e.

7. Claim 15

a. 15[pre]: A receiving device comprising a decoder for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, the decoder comprising:

148. If limiting, in my opinion, Au discloses or suggests the 15[pre]. As discussed in Section IX.C.6.a for 11[pre], it is my opinion that Au discloses or suggests "A decoder for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices." 15[pre] adds only a "receiving device" comprising such a decoder. It is my opinion that this additional requirement is also disclosed by Au, which discloses the

exchange of video data, including decoding, in devices such as "personal computers, wireless devices, surveillance systems, video conferencing systems and set-top boxes." Au, 7:17–22. A POSITA would have understood those devices to be "receiving devices" that comprise a decoder because such devices receive and store data, and a POSITA, in my opinion, would have also understood that decoding in Au is performed in such devices because the received data is what is decoded.

b. 15[a]: a processor configured for:

149. In my opinion, Au discloses or suggests 15[a] for substantially the same reasons as 11[a]. *See* Section IX.C.6.b.

c. 15[b]: recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set;

150. In my opinion, Au discloses or suggests 15[b] for substantially the same reasons as 11[b]. *See* Section IX.C.6.c.

d. 15[c]: recognizing a picture parameter set for a picture and forming at least one first picture parameter value pertaining to a picture using the parameter set;

151. In my opinion, Au discloses or suggests 15[c] for substantially the same reasons as 11[c]. *See* Section IX.C.6.d.

e. 15[d]: forming at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

152. In my opinion, Au discloses or suggests 15[d] for substantially the same reasons as 11[d]. *See* Section IX.C.6.e.

f. 15[e]: using the at least one second picture parameter value in decoding.

153. In my opinion, Au discloses or suggests 15[e] for substantially the same reasons as 11[e]. *See* Section IX.C.6.f.

8. Claim 21

a. 21[pre]: A computer readable storage medium embedded with a computer program comprising programming code for execution on a processor, the programming code for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, and the programming code for:

154. If limiting, in my opinion, Au discloses or suggests the preamble. As discussed in Section IX.C.2.a for 6[pre], Au discloses or suggests "decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices." 21[pre] adds only the requirement of a "computer readable storage medium embedded with a computer program comprising programming code for execution on a processor" and that program is for the "decoding" similarly recited in 6[pre]. Au discloses this addition.

155. As discussed in Section IX.C.6.a for 11[pre], Au's decoding is done by a decoder. Au, 6:54–55 ("FIG. 3 shows further details of a decoder of the system of FIG. 1;"), 10:41–46 ("The decoder **20** (see FIG. **3**) of the receiving participant A,B processes the received bit stream **15** and then reconstructs the coded frame…."), Fig. 3. Au further discloses "the processors **16** coordinate the display of successive frames **22** on the monitors **13**, as the video data **12** is communicated between the participants A, B during a video conference." *Id.*, 7:11–14. And as discussed in Section IX.C.6.b for 11[a], a

POSITA would have understood that Au discloses a processor for decoding video. *Id*. A POSITA would have further understood that such a decoder using a processor would have been implemented with programming code in a computer program stored on a computer readable storage medium.

b. 21[a]: recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set;

156. In my opinion, Au discloses or suggests 21[a] for substantially the same reasons as 11[b]. *See* Section IX.C.6.c.

c. 21[b]: recognizing a picture parameter set for a picture and forming at least one first picture parameter value pertaining to a picture using the parameter set;

157. In my opinion, Au discloses or suggests 21[b] for substantially the same reasons as 6[b]. *See* Section IX.C.2.c.

d. 21[c]: forming at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

158. In my opinion, Au discloses or suggests 21[c] for substantially the same reasons as 6[c]. *See* Section IX.C.2.d.

e. 21[d]: using the at least one second picture parameter value in decoding.

159. In my opinion, Au discloses or suggests 21[d] for substantially the same reasons as 6[d]. *See* Section IX.C.2.e.

9. Claim 23

a. 23[pre]: A decoder for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, the decoder comprising:

160. If limiting, in my opinion, Au discloses or suggests 23[pre] for substantially the same reasons as 11[pre]. *See* Section IX.C.6.a.

b. 23[a]: means for recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set for a picture;

161. In my opinion, Au discloses or suggests 23[a] regardless of whether it is construed as MPF.

162. In my opinion, Au teaches the function as discussed in Section IX.C.6.c for limitation 11[b].

163. If this term is construed as MPF, in my opinion, Au discloses or suggests the structure discussed in Section VIII.A("a decoder as described in the '818 patent specification in figure 4 and equivalents thereof"). As shown below, Figure 4 of the '818 patent discloses a structure that includes a processor, decoding buffer, program code (the decoding buffer and program code both stored in memory).

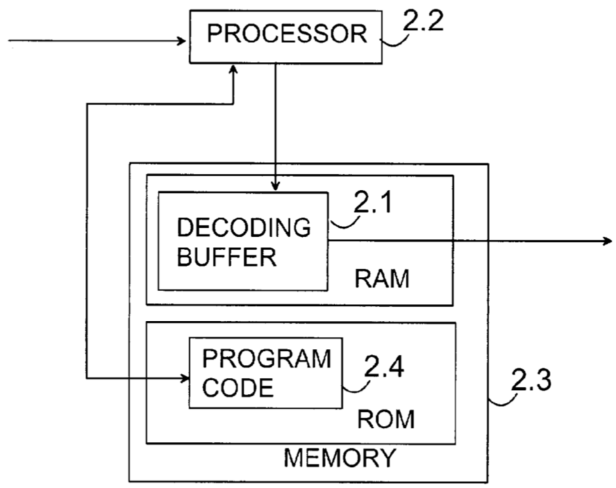


Fig. 4

164. As discussed in Section IX.C.6.c for 11[b], it is my opinion that Au discloses "recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set for a picture." Similarly, as noted for claim limitation 6[a], it is my opinion that Au discloses that this recognition is done "in a decoder." Au further discloses a decoder for converting a bit stream into a series of coded frames. Au, 6:54–55, 10:41–11:25, Fig. 3. As Section IX.C.6.b for 11[a] explains, Au discloses the decoder includes a processor. As disclosed for limitation 21[pre], a POSITA would have understood, or found it obvious, that Au's decoder uses a processor implemented with programming code in a computer program stored on a computer readable storage medium. Au's

decoder also includes a "buffer **500** at the input to the decoder **20**." *Id.*, 11:5–10, 12:20 ("decoder buffer **500**"). Au further discloses that the decoder buffer is stored using memory. *Id.*, 12:20–24 (discussing how decoder buffer affects "memory requirements"). For these reasons, a POSITA, in my opinion, would have understood Au disclosed a decoder with structure the same as, or equivalent to, that of Figure 4 of the '818 patent, and that decoder performs the recited function in 23[a].

c. 23[b]: means for recognizing a picture parameter set and forming at least one first picture parameter value pertaining to a picture using the parameter set;

165. In my opinion, Au discloses or suggests 23[b] regardless of whether it is construed as MPF.

166. In my opinion, Au teaches the function as discussed in Section IX.C.2.c for 6[b].

167. If this term is construed as MPF, in my opinion, Au's decoder discloses or suggests the structure discussed in Section VIII.A("a decoder as described in the '818 patent specification in figure 4 and equivalents thereof") for the same reasons discussed in Section IX.C.9.b for 23[a] above and that decoder performs the recited function in 23[b].

d. 23[c]: means for forming at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

168. In my opinion, Au discloses or suggests 23[c] regardless of whether it is construed as MPF.

169. In my opinion, Au teaches the function as discussed in Section IX.C.2.d for 6[c].

170. If this term is construed as MPF, in my opinion, Au's decoder discloses or suggests the structure discussed in Section VIII.A("a decoder as described in the '818 patent specification in figure 4 and equivalents thereof") for the same reasons discussed in Section IX.C.9.b for 23[a] above and that decoder performs the recited function in 23[c].

e. 23[d]: means for using the at least one second picture parameter value in decoding.

171. In my opinion, Au discloses or suggests 23[d] regardless of whether it is construed as MPF.

172. In my opinion, Au teaches the function as discussed in Section IX.C.2.e for 6[d].

173. If this term is construed as MPF, in my opinion, Au's decoder discloses or suggests the structure discussed in Section VIII.A("a decoder as described in the '818 patent specification in figure 4 and equivalents thereof") for the same reasons discussed in Section IX.C.9.b for 23[a] above and that decoder performs the recited function in 23[d].

D. Ground 4: Obviousness over VCEG-N52 in view of Sriram (Claims 6–9, 11, 15, 21, and 23)

174. It is my opinion that claims 6–9, 11, 15, 21, and 23 of the '818 patent are unpatentable as obvious in view of VCEG-N52 and Sriram, either on their own or in

further combination with the knowledge of a POSITA.

1. VCEG-N52

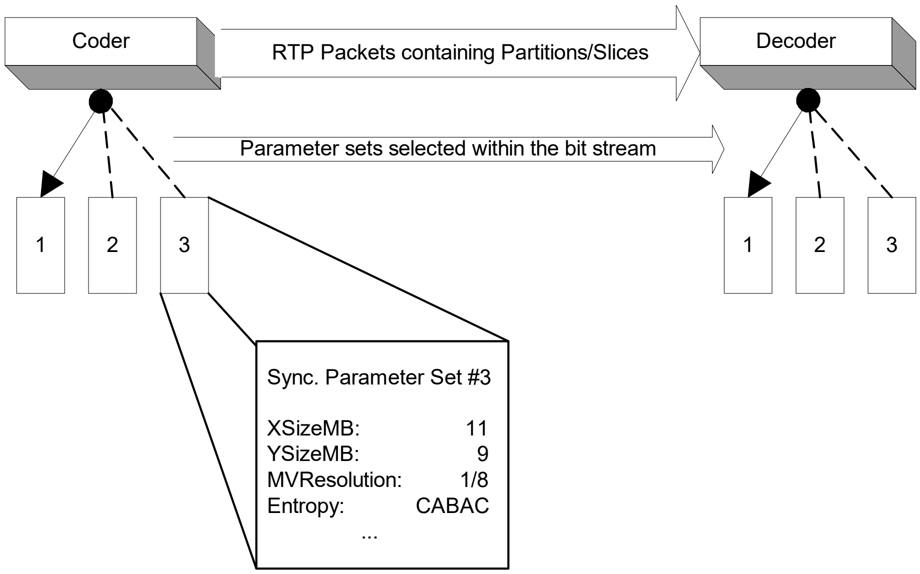
175. It is my understanding that VCEG-N52 qualifies as a printed publication, which the ’818 applicant filed during prosecution and admitted was prior art. ’818FH, 146. VCEG-N52 was publicly available before the earliest priority date on the face of the ’818 patent. Munford, ¶¶18–30; Reader, ¶¶51–55. VCEG-N52 itself indicates it was “Generated: 18 September '01,” for a September 2001 meeting in Santa Barbara, California. I also understand that June Munford and Clifford Reader each provided declarations (EX-1014 and EX-1025, respectively) that described the public availability of this document. Therefore, it is my understanding that VCEG-N52 was publicly available in September 2001—and at least by April 2002—and is prior art under §102(b).

176. VCEG-N52 describes a “concept for the transport of H.26L over IP and H.324 networks” using parameter sets. VCEG-N52, 1. A video coder generates parameter sets and slice headers that indicate the parameter set to be used. *Id.*, 3–4.[4] A video decoder then “maintains a number of parameter set storages” and uses the information from the slice header to determine which parameter set to use for that slice. *Id.*, 4. VCEG-N52 includes a figure, reproduced below,[5] showing the transmission of

---

[4] A video coder is also sometimes referred to as a video encoder.
[5] For clarity, this figure is taken from the Microsoft Word version of VCEG-N52, available at https://www.itu.int/wftp3/av-arch/video-site/0109_San/VCEG-N52.doc. It is substantively identical to the figure in EX-1012.

parameter sets within the bit stream. *Id.*, 3.



177. VCEG-N52 discloses a syntax for transmitting parameter sets, with a parameter set identifier ("ParameterSetNumber"), parameter name ("ParameterName") and parameter value of ("ParameterValue"). *Id.*, 10.

a=H26L:<AttributeValue>, whereby

<AttributeValue> ::= (<ParameterSetNumber>*) <ParameterName> <ParameterValue>

<ParameterSetNumber> ::= int

<ParameterName> ::= (MaxPicID | PixAspectRatio | DisplayWindowOffsetTop | DisplayWindowOffsetBottom | DisplayWindowOffsetRight | DisplayWindowOffsetLeft | XsizeMB | YSizeMB | EntropyCoding | MotionResolution | SliceType)

<ParameterValue> ::= string

178. VCEG-N52 discloses different parameters, which are related to different levels of the video hierarchy (Sequence, Group of Pictures ("GOP"), Picture, Slice). *Id.*, 8. VCEG-N52 explicitly discloses storing a parameter set identifier in the slice header. *Id.*, 6.

ParameterSet	UVLC codeword, symbol number represents the parameter storage location used for the slice.

2. Sriram

179. Sriram discloses a video decoding system with an explicit “design goal” of “efficiency.” Sriram, 1:43. In particular, Sriram takes advantage of the fact that there are “shared parameters in a bit stream.” *Id.*, 1:43–47. Sriram includes an example where “slice headers” can be used to “identify portions of a video frame which can be independently processed.” *Id.*, 1:47–49. As with VCEG-N52’s disclosure of different levels of a video hierarchy, with different parameters at each level, Sriram discloses “several parameters which [are] extracted from the picture, sequence and GOP headers” that are used for decoding a macroblock. *Id.*, 12:64–66, 6:1–5. Sriram defines “[s]everal smaller data structures” containing these parameters, where pointers to these data structures are included in the slice and macroblock structures. *Id.*, 12:66–13:1, 6:19–22. Sriram states as a constraint of the invention that “[a]ll parameter data that is needed at a certain level shall be accessible from the data structure defined for that level.” *Id.*, 6:8–11. For example, Sriram discloses a picture parameter (“Picture Horizontal Size”), which is stored in a picture parameter data structure with a pointer to it from every slice data structure. *Id.*, 13:2–9. Sriram provides an exemplary data structure (“PicParam”), reproduced below, that “contains all the different picture-level parameters that are also needed at the [macroblock] level….” *Id.*, 14:35–54.

```
typedef struct {
        int horizontal__size;
        int vertical__size;
        int code__picture__width;
        int code&__pictureheight;
        int pict type;
        int pictstruct;
        int progframe;
        int prog__seq;
        int chrorna__format;
        int framepred__dct;
        int conceal__my;
        int repeatfirst;
        int topfirst;
        int secondfield;
} PicParam;
```

180. Sriram also discloses a "slice data structure," reproduced below, which contains a pointer ("picparam") with an asterisk indicating a pointer to a PicParam data structure. *Id.*, 15:57–65; Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, ("Kernighan and Ritchie" EX-1024), 8 (describing using the asterisk to declare a pointer variable).

```
typedef struct slice__data {
    SliceStrptr *btstream;
    int MBAstart;
    int MBAstop;
    int VideoType;
    MvecMisc *mvparam;
    PicParam *picparam;
    PicBlockparam *picBlockParm;
    PicFrame    *PicFramePtr;
    vlcptr    *vlcPtr;
} SliceData;
```

Sriram, 13:35–45.

181. Sriram's slice data structure also includes a pointer (as an asterisk) to a picture frame data structure ("PicFrame *PicFramePtr"). *Id.*, 15:57–65. Because Sriram includes parameters within data structures for each level, in my opinion, a POSITA would have understood (or found it obvious) that Sriram includes sequence

parameters in a sequence parameter data structure, and includes GOP parameters in a GOP parameter data structure. *Id.*, 6:8–11. Further, because Sriram's "primary data structures" have pointers to "secondary data structures," in my opinion, a POSITA would have understood (or found it obvious) that the picture parameter data structure includes a pointer to the GOP parameter data structure, and the GOP parameter data structure includes a pointer to the sequence parameter data structure. *Id.*, 6:19–22.

3. Motivation to combine VCEG-N52 and Sriram

182. In my opinion, a POSITA would have been motivated to combine VCEG-N52 and Sriram to extend VCEG-N52 based on Sriram's teachings about efficient data structures and would have had a reasonable expectation of success in doing so for several reasons. First, both references disclose video decoding and processing parameters for video decoding. For example, both references disclose (1) the same video hierarchy of sequence, GOP, picture, slices, and macroblocks; VCEG-N52, 4, 8; Sriram, 3:14–18, 6:4–5, 9:29–30; (2) how to efficiently process the parameters for these levels contained within a bitstream; VCEG-N52, 8; Sriram, 1:43–44; (3) storing parameters together, "parameter set" in VCEG-N52 and data structures in Sriram; VCEG-N52, 3; Sriram, 12:64–13:13, 14:35–54; and (4) identifying the parameter set to use for each slice; VCEG-N52, 6 (slice header includes "ParameterSet" identifier); Sriram, 15:56–67 ("SliceData" data structure includes "PicParam *picparam").

183. In my opinion, because VCEG-N52 and Sriram both focus on efficiently processing parameters, a POSITA would have been motivated to improve VCEG-N52's

parameter set concept based on Sriram's teaching of "[s]everal smaller data structures" for holding the "several parameters" extracted from the "picture, sequence and GOP headers…." VCEG-N52, 3; Sriram, 12:64–13:13. Put differently, in my opinion, a POSITA would be motivated to break up VCEG-N52's parameter set into multiple parameter sets for each level of the video hierarchy. In my opinion, doing so would have contributed to the shared goal of VCEG-N52 and Sriram of increasing bandwidth. Sriram, 6:29–33; VCEG-N52, 8 (noting the advantage of "Good compression" "once the parameters are available at the decoder"). In the combination, each level of parameters would need to be transmitted only once, as part of the related parameter set, as opposed to being included in every parameter set. For example, consider the situation where a picture parameter changes under the disclosure of VCEG-N52, without the teachings of Sriram. In such a case, in my opinion, the system would need to transmit all of the parameters (including those at the GOP and sequence levels) even though only one parameter at the picture level had changed. Based on the disclosure of VCEG-N52, as shown below, this would require six integer variables and one floating point variable. VCEG-N52, 8.

Sequence Parameters

int MaxPicId	Maximum number for the PicId Calculation. Is typically smaller than or equal to the number of reference pictures.
float PixAspectRatio	Pixel geometry, calculated by SizeX / SizeY, 0.0 indicates "unspecified"

Group Of Picture Parameters

A Group Of Picture compromises a random access unit, normally starting with an Intra picture (what about SP pictures? Need to discuss!).

int DisplayWindowOffsetTop	The visible display window (may be smaller than the size in MBs of the coded picture
int DisplayWindowOffsetBottom	
int DisplayWindowOffsetRight	
int DisplayWindowOffsetLeft	
int XsizeMB	Picture size, x dimension, in macroblocks
int YSizeMB	

184. Based on the teachings of Sriram to split the parameter sets up, however, in my opinion only a single variable would need to be sent along with the updated picture parameter, to identify the GOP parameter set previously sent. The savings come from the fact that the amount of data required to identify a parameter set is less than the amount of data required for all of the redundant parameters. A POSITA would have used Sriram's parameter data structures not just for processing on the decoding end of the system, but also for transmission, to get additional bandwidth savings during transmission.

185. In my opinion, extending VCEG-N52's parameter set concept with Sriram's multiple parameter sets would involve only routine engineering tasks that would yield predictable results. In particular, this combination would have been accomplished using well-known software engineering techniques applied to the disclosures of VCEG-N52 and Sriram, and would have been well within the capabilities

of a POSITA to program. VCEG-N52's parameter set syntax includes parameters for each level of the video hierarchy, as shown below (sequence parameters highlighted dark red, group of picture parameters highlighted light red, picture parameters highlighted green).

<AttributeValue> ::= (<ParameterSetNumber>*) <ParameterName> <ParameterValue>

<ParameterSetNumber> ::= int

<ParameterName> ::= (MaxPicID | PixAspectRatio | DisplayWindowOffsetTop | DisplayWindowOffsetBottom | DisplayWindowOffsetRight | DisplayWindowOffsetLeft | XsizeMB | YSizeMB | EntropyCoding | MotionResolution | SliceType)

<ParameterValue> ::= string

186. In my opinion, separating these parameters into different parameter sets would have been a routine software engineering task with expected results. Further, in order to encapsulate all the parameters used in decoding of a picture in a single data structure, a POSITA would have added the items included in Sriram's PicParam data structure to the picture parameter set. Sriram, 14:35–54. In addition, because Sriram's "primary data structures" have pointers to "secondary data structures," a POSITA would have included, within each parameter set, a pointer to the next higher parameter set (i.e., picture parameter set includes pointer to GOP parameter set ("GOPParameterSetIdentifier"); GOP parameter set includes pointer to sequence parameter set ("SeqParameterSetIdentifier")). *Id.*, 6:19–22. Such a reference between levels of the video hierarchy is also taught by Sriram's slice data structure, which includes a pointer to the picture parameter data structure. *Id.*, 15:56–67 ("PicParam

*picparam”). The following is an example of the syntaxes for the sequence and picture parameter sets in accordance with these improvements taught by Sriram.

<SeqAttributeValue> ::= (<SeqParameterSetNumber>*) <SeqParameterName> <SeqParameterValue>

<SeqParameterSetNumber> ::= int

<SeqParameterName> ::= (MaxPicID | PixAspectRatio)

<SeqParameterValue> ::= string

<GOPAttributeValue> ::= (<GOPParameterSetNumber>*) <GOPParameterName> <GOPParameterValue>

<GOPParameterSetNumber> ::= int

<GOPParameterName> ::= (SeqParameterSetIdentifier | DisplayWindowOffsetTop | DisplayWindowOffsetBottom | DisplayWindowOffsetRight | DisplayWindowOffsetLeft | XsizeMB | YSizeMB)

<GOPParameterValue> ::= string

<PicAttributeValue> ::= (<PicParameterSetNumber>*) <PicParameterName> <PicParameterValue>

<PicParameterSetNumber> ::= int

<PicParameterName> ::= (GOPParameterSetIdentifier | horizontal_size | vertical_size | code_picture_width | code&_pictureheight | pict_type | pictstruct | progframe | prog_seq | chroma_format | framepred_dct | conceal_my | repeatfirst | topfirst | secondfield | EntropyCoding | MotionResolution)

<PicParameterValue> ::= string

187. In my opinion, a POSITA would have recognized the above syntax would improve on VCEG-N52’s parameter set concept by using the teachings of Sriram, and its implementation would require only routine well known and well established software engineering techniques.

4. Claim 6

a. 6[pre]: A method for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, the method comprising:

188. If limiting, in my opinion, the combination of VCEG-N52 and Sriram teaches or suggests the preamble 6[pre]. VCEG-N52 and Sriram each disclose a video decoder. *See, e.g.*, VCEG-N52, 4; Sriram, 4:11–12, Figure 2. In my opinion, a POSITA would have understood that the combination would also include a decoder.

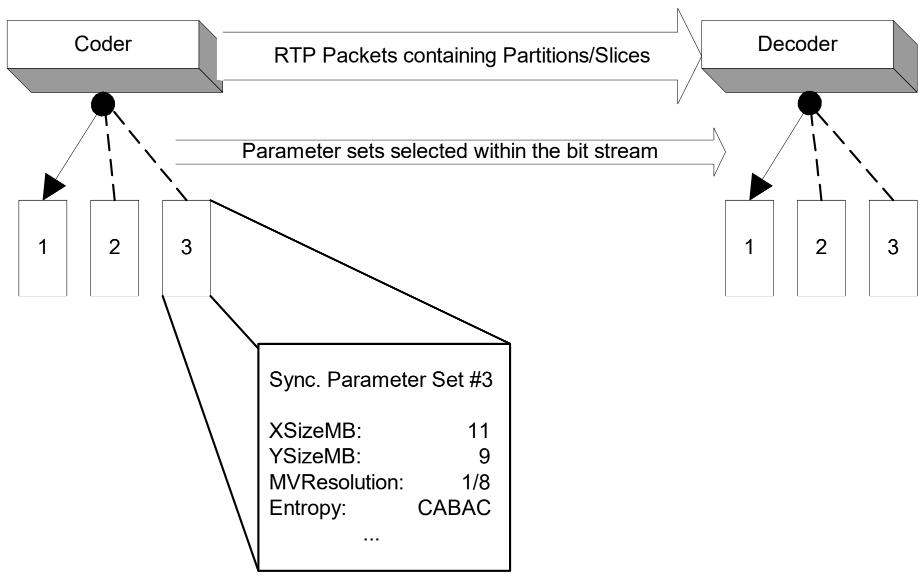
189. As to "decoding sequences of pictures from a bitstream" and "each picture comprises information of one or more slices," both VCEG-N52 and Sriram disclose decoding pictures comprised of slices from a bitstream. VCEG-N52, 5, 6, 12 (describing the bitstream), 1 ("RTP [Real-time Transit Procotol] stream" carried slices, and pictures are divided into slices); Sriram, 6:1–5 (describing a "hierarchy of data structures," with "slice" coming just under "picture"), 4:59–5:14 (describing an incoming bitstream which is decoded into video frames).

190. As to "parameters are defined in a parameter set," as admitted by the '818 applicant, VCEG-N52 discloses the "Parameter Set Concept." EX-1001, 2:49–58; EX-1002, 142; VCEG-N52, 3. As discussed above in Section IX.D.3, in my opinion, a POSITA would have been motivated to combine VCEG-N52's with Sriram's teaching of different data structures for different levels of the video hierarchy, and doing so would involve only routine software engineering tasks with predictable results.

b. 6[a]: recognizing, in a decoder, a sequence parameter set and forming at least one sequence parameter pertaining to a sequence using the parameter set;

191. In my opinion, the combination of VCEG-N52 and Sriram teaches 6[a]. As described above, the applicant for the '818 patent acknowledged that VCEG-N52 disclosed the parameter set concept. As further described above, a POSITA would have been motivated to extend the parameter set concept disclosed by VCEG-N52 to each level of the video hierarchy, including the sequence and GOP levels, based on the teachings of Sriram. In particular, Sriram discloses that "there are several parameters which were extracted from the picture, ***sequence*** and ***GOP*** headers which are needed for processing an individual MB [macroblock]" (Sriram, 12:64–66 (emphasis added)) and "[s]everal smaller data structures are defined and a pointer to these data structures is included in the Slice and MB data structures" (*id.*, 12:66–13:1). Sriram's example picture parameter data structure is non-limiting (*id.*, 13:2–9; 17:55–60), and a POSITA would have understood (or found it obvious) that Sriram at least suggests a sequence parameter data structure storing sequence parameters and a GOP parameter data structure storing GOP parameters, with the combination including at least VCEG-N52's "Sequence Parameters" and "Group of Picture Parameters." VCEG-N52, 8. A POSITA would have been motivated to make this combination so that sequence and GOP parameters would not need to be repeated in each picture parameter data structure to promote Sriram's stated goal of reducing "data bandwidth." Sriram, 6:29–33, 13:11–13.

192. In my opinion, a POSITA would have understood that "recognizing, in the decoder" as recited in 6[a] encompasses a decoder parsing the incoming bitstream and identifying a "sequence parameter set." VCEG-N52 performs such "recognizing." Figure 1 of VCEG-N52, reproduced below, shows "Parameter sets selected within the bit stream," sent from the "Coder" to the "Decoder."



193. VCEG-N52 describes "Transmission of Parameter sets," including how the information is formatted for the decoder to recognize the parameter set. VCEG-N52, 9–11. This includes identifying a parameter set ("ParameterSetNumber") and the parameters within that set ("ParameterName") and their accompanying values ("ParameterValue"). *Id.*, 10. The relevant portion of this format is shown below.

<ParameterSetNumber> ::= int

<ParameterName> ::= (MaxPicID | PixAspectRatio | DisplayWindowOffsetTop | DisplayWindowOffsetBottom | DisplayWindowOffsetRight | DisplayWindowOffsetLeft | XsizeMB | YSizeMB | EntropyCoding | MotionResolution | SliceType)

<ParameterValue> ::= string

194. As described above in Section IX.D.3, a POSITA would have been motivated to use the parameter set transmission taught by VCEG-N52 to send from the coder, and receive by the decoder, the parameters related to a sequence and GOP. The extended format would resemble the following:

<SeqAttributeValue> ::= (<SeqParameterSetNumber>*) <SeqParameterName> <SeqParameterValue>

<SeqParameterSetNumber> ::= int

<SeqParameterName> ::= (MaxPicID | PixAspectRatio)

<SeqParameterValue> ::= string

<GOPAttributeValue> ::= (<GOPParameterSetNumber>*) <GOPParameterName> <GOPParameterValue>

<GOPParameterSetNumber> ::= int

<GOPParameterName> ::= (SeqParameterSetIdentifier | DisplayWindowOffsetTop | DisplayWindowOffsetBottom | DisplayWindowOffsetRight | DisplayWindowOffsetLeft | XsizeMB | YSizeMB)

<GOPParameterValue> ::= string

195. Each of the GOP parameter set and sequence parameter set is "a sequence parameter set" as recited by this limitation. A POSITA would have understood that a group of pictures is a sequence of pictures. *See* Watkinson, 11 (describing GOP as "An *I* picture together with all of the pictures before the next *I* picture…."). The '818 patent itself refers to an "independent GOP parameter set or a sequence parameter set" interchangeably. EX-1001, 7:20–21, 7:54–57 ("a picture parameter set 104 refers to both an independent GOP parameter set 105 and a presentation parameter set 106, or to a sequence parameter set"), 8:45–54 ("Picture Parameter Set" including only reference to "an independent GOP parameter set"), Fig. 1 (including "Independent GOP Parameter Set" but not a "Sequence Parameter Set"). Therefore, a POSITA would have

recognized that both the GOP parameter set and sequence parameter set of the combination of VCEG-N52 and Sriram would meet the "sequence parameter set" limitation.

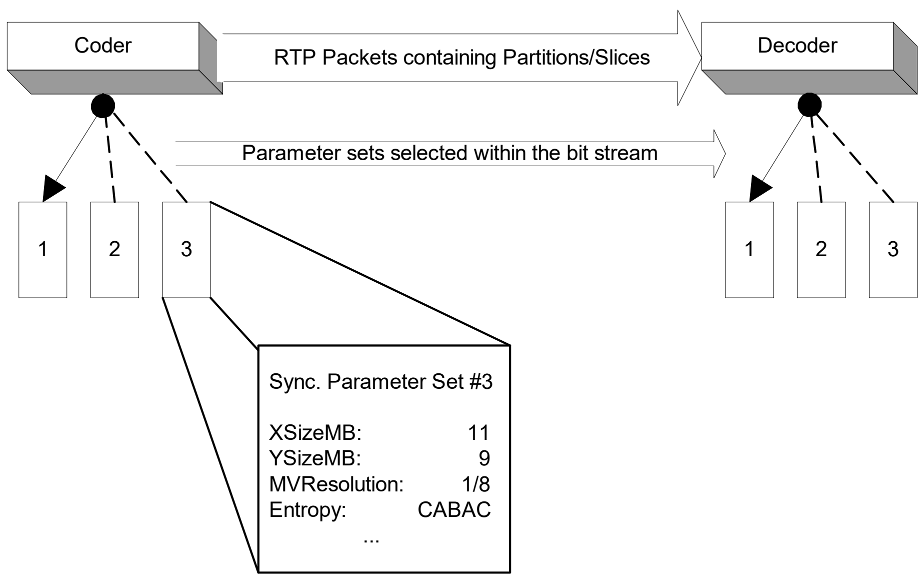
196. In my opinion, a POSITA would have further understood that the combination of VCEG-N52 and Sriram includes "forming" (e.g., creating in local memory) a sequence parameter. In particular, as shown in the data format above, a syntax for the combination of VCEG-N52 and Sriram includes variables "SeqParameterValue" and "GOPParameterValue" that contain values of a sequence parameter included in the sequence parameter set and GOP parameter set, respectively. In my opinion, a POSITA would have understood that these variables and their associated values would be extracted by the decoder and stored in local memory for later use.

c. *6[b]: recognizing, in the decoder, a picture parameter set and forming at least one first picture parameter value pertaining to a picture using the parameter set;*

197. In my opinion, the combination of VCEG-N52 and Sriram teaches 6[b]. VCEG-N52 discloses the parameter set concept, and Sriram discloses a picture parameter data structure. VCEG-N52, 3; Sriram, 15:57–65. Sriram's picture parameter data structure includes multiple picture parameters, each of which teaches "at least one first picture parameter value,", including, among other things, "horizontal_size," "vertical_size," and "chroma_format." Sriram, 14:35–54. As discussed above in Section IX.D.3, a POSITA would have been motivated to combine VCEG-N52 with

the picture parameters stored in Sriram's picture parameter data structure.

198. A POSITA would have understood that "recognizing, in the decoder" as recited in 6[b] encompasses a decoder parsing the incoming bitstream and identifying a "picture parameter set." VCEG-N52 performs such "recognizing." Figure 1 of VCEG-N52, reproduced below, shows "Parameter sets selected within the bit stream," sent from the "Coder" to the "Decoder."



199. VCEG-N52 also describes "Transmission of Parameter sets," including how the information is formatted for the decoder to recognize the parameter set. VCEG-N52, 9–11. This includes identifying a parameter set ("ParameterSetNumber") and the parameters within that set ("ParameterName") and their accompanying values ("ParameterValue"). *Id.*, 10. The relevant portion of this format is shown below.

<ParameterSetNumber> ::= int

<ParameterName> ::= (MaxPicID | PixAspectRatio | DisplayWindowOffsetTop | DisplayWindowOffsetBottom | DisplayWindowOffsetRight | DisplayWindowOffsetLeft | XsizeMB | YSizeMB | EntropyCoding | MotionResolution | SliceType)

<ParameterValue> ::= string

200. In the combination of VCEG-N52 and Sriram, the picture parameter set would include the parameters stored in Sriram's picture parameter data structure ("PicParam," shown below). Sriram, 14:35–54.

```
typedef struct {
        int horizontal_size;
        int vertical_size;
        int code_picture_width;
        int code&_pictureheight;
        int pict type;
        int pictstruct;
        int progframe;
        int prog_seq;
        int chrorna_format;
        int framepred_dct;
        int conceal_my;
        int repeatfirst;
        int topfirst;
        int secondfield;
} PicParam;
```

201. Further, as described above in Section IX.D.3, because Sriram's "primary data structures" have pointers to "secondary data structures," a POSITA would have included, within the picture parameter set, a pointer to the sequence parameter set.*Id.*, 6:19–22. This would have required only routine well known and well established software engineering techniques. The syntax for the picture parameter set would have resembled the following.

<PicAttributeValue> ::= (<PicParameterSetNumber>*) <PicParameterName> <PicParameterValue>

<PicParameterSetNumber> ::= int

<PicParameterName> ::= (GOPParameterSetIdentifier | horizontal_size | vertical_size | code_picture_width | code&_pictureheight | pict_type | pictstruct | progframe | prog_seq | chroma_format | framepred_dct | conceal_my

| repeatfirst | topfirst | secondfieldEntropyCoding | MotionResolution)

<PicParameterValue> ::= string

202. In my opinion, a POSITA would have further understood that the combination of VCEG-N52 and Sriram includes “forming” (e.g., creating in local memory) a picture parameter. In particular, as shown in the data format above, “PicParameterValue” is a variable that contains the value of a picture parameter included in the picture parameter set. In my opinion, a POSITA would have understood that such a variable and its associated value would be extracted by the decoder and stored in local memory for later use.

d. 6[c]: forming, in the decoder, at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

203. In my opinion, the combination of VCEG-N52 and Sriram teaches 6[c]. Both VCEG-N52 and Sriram disclose sending slice headers from the coder to the decoder, and extracting information from such slice headers. VCEG-N52, 4; Sriram, 1:47–49, 9:26–29.

204. In my opinion, a POSITA would have understood Sriram’s slice data structure (“SliceData”) includes both slice data and slice header information (e.g., a pointer to a data bitstream (“btstream”), as well as variables that relate to the data (e.g., “VideoType,” “PicFramePtr”)). Sriram, 15:9-10, 15:52–66. While Sriram does not provide source code for the “PicFrame” structure to which PicFramePtr points, in my opinion, a POSITA would have understood from the disclosure of Sriram what such a

structure would be. For example, Sriram discloses structures written in the C programming language, using the "typedef struct" keywords. *See, e.g.*, *id.*, 13:34–47, 14:26–34, 14:40–53, 14:63–15:7. A POSITA would have understood that the PicFrame would also have been defined using "typedef struct." Kernighan and Ritchie, 10–12. Using standard programming techniques, including those described in Kernighan and Ritchie, a POSITA would have included in such a structure the proper fields for all of the data that describes the Picture/Frame. *Id.* ("Structures help to organize complicated data, particularly in large programs, because they permit a group of related variables to be treated as a unit instead of separate entities."). As stated in Sriram, this would have included "the starting address of all the frames that are needed" for decoding. Sriram, 15:9–10. A POSITA would have understood that a proper reference to the frame being decoded would be included in "all the frames that are needed" for decoding because (1) a POSITA would have needed to know where to place the decoded data and (2) data from the frame being decoded may be needed for INTRA and/or INTER coding. As a further use of data encapsulation, in my opinion, a POSITA would have included the identifier of the frame being decoded (i.e., the "frame number") in the PicFrame. Because each of the addresses of the frame being decoded and the frame number of the frame being decoded are defined by the frame being decoded, a POSITA would have understood these variables are picture parameters, and that all of the slice data structures for a particular picture would have the same value for these parameters (i.e., "one second picture parameter value using information of a slice header," which "remain[s] unchanged at least in all slice headers of one picture"). In the combination of VCEG-N52 and Sriram, the slice data structure and the PicFrame data structure would have been stored in the decoder ("forming, in the decoder, at least one second picture parameter value using information of a slice header"). Sriram, 4:59–62, 5:51–6:43.

205. Alternatively, in my opinion, a POSITA would have been motivated to include the identifier for the frame as a field within the slice data structure. VCEG-N52 discloses a PictureID field, which "represents the PictureID of the picture the slice belongs to" as a variable "traditionally associated with the Slice header…." VCEG-N52, 4–5. A POSITA would have understood that for non-interlaced video, the majority of digital video, a picture and a frame are interchangeable terms. Similarly, Sriram states that its invention is applicable to H.263. Sriram, 5:38–45. The H.263 standard describes a Group of Blocks (GOB) Frame ID, which "shall have the same value in every GOB (or slice) header of a given picture." EX-1022, 19, 32–33. Further, a POSITA would have understood picture order count, another form of frame identifier (and in some cases the same as frame number), may be included in the slice header. Watkinson, 14. Given these teachings, in the combination of VCEG-N52 and Sriram, a POSITA would have included the frame number ("one second picture parameter value using information of a slice header," which "remain[s] unchanged at least in all slice headers of one picture") in the slice data structure, which contains slice header information ("forming, in the decoder, at least one second picture parameter value using information of a slice header").

e. 6[d]: using, in the decoder, the at least one second picture parameter value in decoding.

206. In my opinion, the combination of VCEG-N52 and Sriram teaches 6[d]. As described above, the frame number value, stored in the slice data structure and/or the PicFrame data structure is "at least one second picture parameter value." A POSITA would have understood, or found it obvious, that the frame number is used in the decoder, because it indicates to the decoder in which frame to decode each slice. *See* Watkinson, 14 ("The frame number is a count which increments in the order of decoding…."). Therefore, the value of the frame number is "the at least one second picture parameter value in decoding."

5. Claim 7: The method according to claim 6, wherein the slice header is included with a reference to a picture parameter set.

207. It is my opinion that VCEG-N52 and Sriram teach claim 7. Claim 7 depends from claim 6, and Section IX.D.4 explains that the combination of VCEG-N52 and Sriram teaches the "method according to claim 6." As discussed in Section IX.D.4.d for 6[c], that combination discloses that Sriram's slice data structure includes slice header information. That slice data structure includes a pointer to a picture parameter data structure. Sriram, 15:52–66 ("PicParam *picparam"). Further, Sriram discloses copying information from headers from all levels, including from the picture level to the slice data structure. *Id.*, 9:57–61. A POSITA would have understood this requires the slice data structure to have a reference to the picture header, including the parameters for that picture. As discussed in Section IX.D.4.c, in the combination of

VCEG-N52 and Sriram, the picture parameter data structure would become a picture parameter set. As such, a POSITA would have understood the "picparam" variable within Sriram's slice data structure ("slice header") to be a reference to the picture parameter set of the combined VCEG-N52 and Sriram. That reference could be a pointer as in Sriram ("picparam") or an integer reference of the picture parameter set ("PicParameterSetNumber").

6. Claim 8: The method according to claim 6, wherein the picture parameter set is included with a reference to a sequence parameter set.

208. It is my opinion that VCEG-N52 and Sriram teach claim 8. Claim 8 depends from claim 6, and Section IX.D.4 explains that combination of VCEG-N52 and Sriram teaches the "method according to claim 6." As discussed in Sections IX.D.3 regarding motivation to combine VCEG-N52 and Sriram and IX.D.4.c for 6[b], in the combination of VCEG-N52 and Sriram, a POSITA would have included, within the picture parameter set, a pointer to the sequence parameter set in keeping with Sriram's disclosure of "primary data structures" having pointers to "secondary data structures." Sriram, 6:19–22. This is included as "GOPParameterSetIdentifier," in the format extended from VCEG-N42 by the teachings of Sriram (emphasis added):

<PicAttributeValue> ::= (<PicParameterSetNumber>*) <PicParameterName> <PicParameterValue>

<PicParameterSetNumber> ::= int

<PicParameterName> ::= (**GOPParameterSetIdentifier** | horizontal_size | vertical_size | code_picture_width | code&_pictureheight | pict_type | pictstruct | progframe | prog_seq | chroma_format | framepred_dct | conceal_my | repeatfirst | topfirst | secondfieldEntropyCoding | MotionResolution)

<PicParameterValue> ::= string

209. In my opinion, "GOPParameterSetIdentifier" is a reference to a GOP parameter set. As discussed above in Section IX.D.4.b, a POSITA would have understood a GOP parameter set to be a "sequence parameter set," as recited in the claims. Thus, the picture parameter set of the combination of VCEG-N52 and Sriram includes a reference to a sequence parameter set, as required by this claim.

210. If a GOP parameter set is not a "sequence parameter set," as recited in the claims, in my opinion, the combination still teaches claim 8. As discussed above in Section IX.D.3, in the combination of VCEG-N52 and Sriram, the parameter set at each level includes a reference to the parameter set at the level above it. In other words, each picture parameter set has a reference ("GOPParameterSetIdentifier") to the GOP parameter set above it, and each GOP parameter set has a reference ("SeqParameterSetIdentifier") to the sequence parameter set above it. There is, therefore, an indirect reference from every picture parameter set to the sequence parameter set two levels above it. In my opinion, a POSITA would have understood that this claim's requirement of including "a reference to a sequence parameter set" would include such an indirect reference.

211. If a GOP parameter set is not a "sequence parameter set," as recited in the claims, and claim 8 requires a direct reference from the picture parameter set to a sequence parameter set, in my opinion, a POSITA would have been motivated to modify the combination of VCEG-N52 and Sriram to include a direct reference. Sriram teaches that "[h]aving two levels of indirection may not be desirable for some parameters which

are extensively used by all the processing blocks." Sriram, 13:13–15. Accordingly, a POSITA would be motivated to either (1) combine the GOP parameter set and the sequence parameter set (i.e., including GOP parameters and sequence parameters in the same parameter set), or (2) include a direct reference from the picture parameter set to both the GOP parameter set and the sequence parameter set. In either case, the combination would include a direct reference from the picture parameter set to a sequence parameter set.

7. Claim 9: The method according to claim 6, wherein the second picture parameter value is selected from a group of information of a picture order count and a frame number.

212. It is my opinion that VCEG-N52 and Sriram teach claim 9. Claim 9 depends from claim 6, and Section IX.D.4 explains that the combination of VCEG-N52 and Sriram teaches the "method according to claim 6." As discussed in Sections IX.D.4.d and IX.D.4.e, in the combination of VCEG-N52 and Sriram, the "second picture parameter value" is the value of the frame number, contained in Sriram's slice data structure and/or the PicFrame data structure, as required by this claim. Sriram, 15:52–66. As described in Sriram, PicFrame, which PicFramePtr points, "defines the starting address of all the frames that are needed for MPEG-2 decoding." *Id.*, 15:9–10. Sriram also discloses that "decoded frames may have to be re-ordered before they can be sent to the display processor," and may be stored in a "video buffer." *Id.*, 5:9–14. In my opinion, a POSITA would have understood this would require being able to identify individual frames, by number or otherwise.

8. Claim 11

a. 11[pre]: A decoder for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, the decoder comprising:

213. If limiting, in my opinion, the combination of VCEG-N52 and Sriram teaches the preamble. As discussed in IX.D.4.a for 6[pre], the combination teaches "decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices." Both VCEG-N52 and Sriram disclose that such decoding is done by a decoder. VCEG-N52, 4; Sriram, 4:11–12, Figure 2.

b. 11[a]: a processor configured for:

214. In my opinion, the combination of VCEG-N52 and Sriram teaches 11[a].

215. VCEG-N52 discloses a coder and decoder connected via an IP network. VCEG-N52, 1, 2. A POSITA would have understood that such a decoder would have been implemented using a processor.

216. Sriram discloses a central processing unit that reads and executes a number of "executable programs utilized to implement the video decoding processes of the invention." Sriram, 4:34–58, Figs. 1, 2. These executable programs include a syntax analyzer, monitor processor, sub-processors, synchronous/synchronization processor, and display processor. *Id.* The syntax analyzer program "extract[s] information from the video bit stream." *Id.*, 5:17–19.

217. In the combination of VCEG-N52 and Sriram, in my opinion, a POSITA

would have understood a processor performs the steps in 11[b]-11[e] as discussed below.

c. *11[b]: recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set for a picture;*

218. As discussed in IX.D.4.b for 6[a], in my opinion, the combination of VCEG-N52 and Sriram teaches "recognizing, in a decoder, a sequence parameter set and forming at least one sequence parameter pertaining to a sequence using the parameter set." 11[b] is substantially similar to 6[a] except 11[b] recites "recognizing a sequence parameter set <u>for a sequence of pictures</u> [and] using the parameter set <u>for a picture</u>." These additions in 11[b] are also taught by the combination of VCEG-N52 and Sriram. Both VCEG-N52 and Sriram disclose a hierarchy of syntax layers, with the level above picture being a group of pictures. VCEG-N52, 8; Sriram, 6:1–7. In my opinion, a POSITA would have understood that a group of pictures is a sequence of pictures. Further, a POSITA would have understood that the values stored in the GOP parameter set, such as "XsizeMB" and "YSizeMB," which together provide the dimensions of each picture within the GOP, in macroblocks, would be used for "a picture" (i.e., every picture within the sequence) as required by 11[b]. VCEG-N52, 8.

d. *11[c]: recognizing a picture parameter set and forming at least one first picture parameter value pertaining to a picture using the parameter set;*

219. In my opinion, the combination of VCEG-N52 and Sriram teaches 11[c] for substantially the same reasons as 6[b]. *See* Section IX.D.4.c.

e. 11[d]: forming at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

220. In my opinion, the combination of VCEG-N52 and Sriram teaches 11[d] for substantially the same reasons as 6[c]. See Section IX.D.4.d.

f. 11[e]: using the at least one second picture parameter value in decoding.

221. In my opinion, the combination of VCEG-N52 and Sriram teaches 11[e] for substantially the same reasons as 6[d]. *See* Section IX.D.4.e.

9. Claim 15

a. 15[pre]: A receiving device comprising a decoder for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, the decoder comprising:

222. If limiting, in my opinion, the combination of VCEG-N52 and Sriram teaches 15[pre]. As discussed in Section IX.D.8.a for 11[pre], in my opinion, this combination teaches: "A decoder for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices." 15[pre] adds only a "receiving device" comprising such a decoder. Both VCEG-N52 and Sriram disclose that the decoder is included within a "receiving device." VCEG-N52 discloses the use of "H.324 terminals" and "IP based terminals" that contain a decoder. VCEG-N52, 1, 3; *see also id.*, 11 (referring to "A Videoconferencing terminal"). Similarly, Sriram describes the invention as being used in workstations, and the decoder as being part of a system that is also connected

to input/output devices, including a mouse, keyboard, computer monitor, and printer. Sriram, 3:54–59, 4:34–41, 17:36–39 ("In particular, the invention is advantageously exploited in emerging servers and workstations that provide HDTV video decoders using symmetric multiprocessing techniques.").

b. 15[a]: a processor configured for:

223. In my opinion, the combination of VCEG-N52 and Sriram teaches 15[a] for substantially the same reasons as 11[a]. *See* Section IX.D.8.b.

c. 15[b]: recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set;

224. In my opinion, the combination of VCEG-N52 and Sriram teaches 15[b] for substantially the same reasons as 11[b]. *See* Section IX.D.8.c.

d. 15[c]: recognizing a picture parameter set for a picture and forming at least one first picture parameter value pertaining to a picture using the parameter set;

225. In my opinion, the combination of VCEG-N52 and Sriram teaches 15[c] for substantially the same reasons as 11[c]. *See* Section IX.D.8.d.

e. 15[d]: forming at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

226. In my opinion, the combination of VCEG-N52 and Sriram teaches 15[d] for substantially the same reasons as 11[d]. *See* Section IX.D.8.e.

f. 15[e]: using the at least one second picture parameter value in decoding.

227. In my opinion, the combination of VCEG-N52 and Sriram teaches 15[e]

substantially the same reasons as 11[e]. *See* Section IX.D.8.f.

10. Claim 21

a. *21[pre]: A computer readable storage medium embedded with a computer program comprising programming code for execution on a processor, the programming code for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, and the programming code for:*

228. If limiting, in my opinion, the combination of VCEG-N52 and Sriram teaches the preamble. As discussed in Section IX.D.4.a for 6[pre], this combination teaches "decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices." Section IX.D.8.b for 11[a] explains this combination teaches a processor configured to perform the recited steps that follow. Beyond 6[pre] and 11[a], 21[pre] adds only the requirement of a "computer readable storage medium embedded with a computer program comprising programming code for execution on a processor" and that program is for the "decoding" similarly recited in 6[pre]. In my opinion, the combination of VCEG-N52 and Sriram discloses this addition.

229. As discussed in Section IX.D.8.a for 11[pre], both VCEG-N52 and Sriram disclose decoding done by a decoder. VCEG-N52, 4; Sriram, 4:11–12, Figure 2. And as discussed in Section IX.D.8.b for 11[a], in my opinion, a POSITA would have understood that both references use a processor for decoding a video. VCEG-N52, 1, 2; Sriram, 4:34–58, Figs. 1, 2.

230. In my opinion, a POSITA would have understood that the combination's

decoder using a processor would have been implemented with programming code in a computer program stored on a computer readable storage medium. Sriram describes "system memory" (a computer readable storage medium) that "stores a set of executable programs" (programming code) "utilized to implement the video decoding process of the invention." Sriram, 4:34–46. As described in Sriram and shown in Figure 1, that system memory is connected to the central processing unit via a system bus. *Id.*, 4:34–38, Fig. 1. A POSITA would have understood that the connection between the central processing unit and the system memory would be such that the central processing unit (a processor) could access and execute the programs stored in system memory.

b. 21[a]: recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set;

231. In my opinion, the combination of VCEG-N52 and Sriram teaches 21[a] for substantially the same reasons as 11[b]. *See* Section IX.D.8.c.

c. 21[b]: recognizing a picture parameter set for a picture and forming at least one first picture parameter value pertaining to a picture using the parameter set;

232. In my opinion, the combination of VCEG-N52 and Sriram teaches 21[b] for substantially the same reasons as 6[b]. *See* Section IX.D.4.c.

d. 21[c]: forming at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

233. In my opinion, the combination of VCEG-N52 and Sriram teaches 21[c] for substantially the same reasons as 6[c]. *See* Section IX.D.4.d.

e. 21[d]: using the at least one second picture parameter value in decoding.

234. In my opinion, the combination of VCEG-N52 and Sriram teaches 21[d] for substantially the same reasons as 6[d]. *See* Section IX.D.4.e.

11. Claim 23

a. 23[pre]: A decoder for decoding sequences of pictures from a bitstream, wherein parameters are defined in a parameter set and each picture comprises information of one or more slices, the decoder comprising:

235. If limiting, in my opinion, the combination of VCEG-N52 and Sriram teaches 23[pre] for substantially the same reasons as 11[pre]. *See* IX.D.8.a.

b. 23[a]: means for recognizing a sequence parameter set for a sequence of pictures and forming at least one sequence parameter pertaining to a sequence using the parameter set for a picture;

236. In my opinion, the combination of VCEG-N52 and Sriram teaches 23[a] regardless of whether it is construed as MPF.

237. In my opinion, the combination of VCEG-N52 and Sriram teaches the function as discussed in IX.D.8.c for 11[b].

238. If that this term is construed as MPF, the combination of VCEG-N52 and Sriram teaches the structure discussed in Section VIII.A("a decoder as described in the '818 patent specification in figure 4 and equivalents thereof"). As shown below, Figure 4 of the '818 patent discloses a structure that includes a processor, decoding buffer, program code (the decoding buffer and program code both stored in memory).

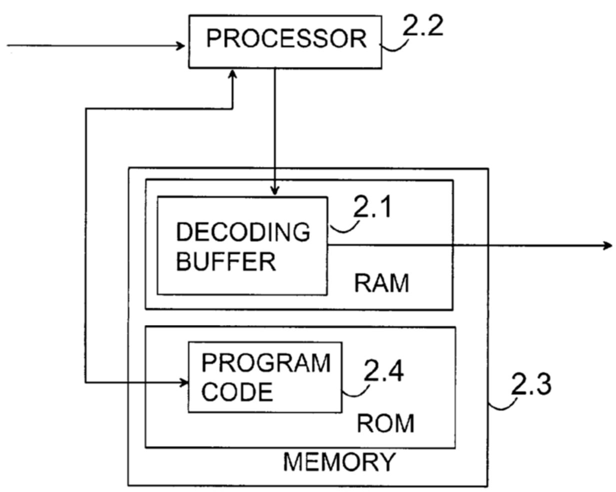


Fig. 4

239. As discussed in Section IX.D.8.c for 11[b], in my opinion, the combination of VCEG-N52 and Sriram discloses that its decoding is performed by a decoder. VCEG-N52, 1, 2; Sriram, 4:34–58, Figs. 1, 2. Section IX.D.8.b for 11[a] explains that this combination teaches a processor configured to perform the recited steps that follow. Section IX.D.10.a for 21[pre] explains that a POSITA would have understood, or found it obvious, that the combination uses a processor implemented with programming code in a computer program stored on a computer readable storage medium. In particular, Sriram discloses storing executable programs in system memory for decoding. Sriram, 4:34–46. The combination of VCEG-N52 and Sriram also teaches using a decoding buffer. VCEG-N52 discloses a "playout buffer" that is filled with packets received by

the decoder. VCEG-N52, 7. Sriram discloses a "video buffer," which contains the "frame outputs" from the "synchronization processor." Sriram, 4:54–58, 5:9–14, Fig. 1. Sriram further teaches "video frames" that are "buffered in memory." *Id.*, 2:52–55, Fig. 2 ("Buffered Video Frames"). Therefore, a POSITA would have understood that the combination of VCEG-N52 and Sriram discloses a decoder with structure the same as, or equivalent to, that of Figure 4 of the '818 patent, and that decoder performs the recited function in 23[a].

c. 23[b]: means for recognizing a picture parameter set and forming at least one first picture parameter value pertaining to a picture using the parameter set;

240. In my opinion, the combination of VCEG-N52 and Sriram teaches 23[b] regardless of whether it is construed as MPF.

241. In my opinion, the combination of VCEG-N52 and Sriram teaches the function as discussed in Section IX.D.4.c for 6[b].

242. If 23[b] is construed as MPF, in my opinion, the combination of VCEG-N52 and Sriram teaches the decoder structure discussed in Section VIII.A ("a decoder as described in the '818 patent specification in figure 4 and equivalents thereof") for the same reasons discussed in Section IX.D.10.b for 23[a] above and that decoder performs the recited function in 23[b].

d. 23[c]: means for forming at least one second picture parameter value using information of a slice header, the at least one second picture parameter value remaining unchanged at least in all slice headers of one picture; and

243. In my opinion, the combination of VCEG-N52 and Sriram teaches 23[c]

regardless of whether it is construed as MPF.

244. In my opinion, the combination of VCEG-N52 and Sriram teaches the function as discussed in IX.D.4.d for 6[c].

245. If 23[c] is construed as MPF, in my opinion, the combination of VCEG-N52 and Sriram teaches the decoder structure discussed in Section VIII.A("a decoder as described in the '818 patent specification in figure 4 and equivalents thereof") for the same reasons discussed in Section IX.D.10.b for 23[a] above and that decoder performs the recited function in 23[c].

e. 23[d]: means for using the at least one second picture parameter value in decoding.

246. In my opinion, the combination of VCEG-N52 and Sriram teaches 23[d] regardless of whether it is construed as MPF.

247. In my opinion, the combination of VCEG-N52 and Sriram teaches the function as discussed in Section IX.D.4.e for 6[d].

248. If term is construed as MPF, in my opinion, the combination of VCEG-N52 and Sriram teaches the decoder structure discussed in Section VIII.A("a decoder as described in the '818 patent specification in figure 4 and equivalents thereof") for the same reasons discussed in Section IX.D.10.b for 23[a] above and that decoder performs the recited function in 23[d].

X. CONCLUSION

249. I currently hold the opinions expressed in this declaration. But my analysis may continue, and I may acquire additional information and/or attain supplemental

insights that may result in added observations.

250. I hereby declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct to the best of my knowledge. I understand that willful false statements are punishable by fine or imprisonment or both. *See* 18 U.S.C. § 1001.

Date: April 1, 2024   By: ______________________

Eli Saber, Ph.D.